
Transparency, Privacy, and Fairness
in
Recommender Systems

CUMULATIVE HABILITATION
FOR THE SCIENTIFIC SUBJECT

APPLIED COMPUTER SCIENCE

SUBMITTED BY

DOMINIK KOWALD

*Institute of Interactive Systems and Data Science
Graz University of Technology*



NOVEMBER 2023

I've got to keep going, be strong. Must be so determined and push myself on.

— Iron Maiden, The Loneliness of the Long Distance Runner —

Dedicated to my wife, Tea.

Overview of Chapters

Chapter 1 - Introduction

This chapter starts with the motivation and the scientific positioning of this habilitation within the broad research field of recommender systems. It also lists and briefly outlines the 17 main publications of this habilitation.

Chapter 2 - Related Work and Background

Chapter 2 briefly discusses the related work and background relevant to this habilitation, namely (i) main concepts of recommender systems, (ii) transparency and cognitive models in recommender systems, (iii) privacy and limited preference information in recommender systems, and (iv) fairness and popularity bias in recommender systems. Additionally, this chapter briefly summarizes the author's own research efforts in relation to the related work.

Chapter 3 - Scientific Contributions

This chapter describes the 7 scientific contributions of this habilitation, which are (i) using cognitive models for a transparent design and implementation process of recommender systems, (ii) illustrating to what extent components of the cognitive model ACT-R contribute to recommendations, (iii) addressing limited user preference information in cold-start and session-based recommendation settings, (iv) addressing users' privacy constraints and the trade-off between accuracy and privacy in recommendations, (v) measuring popularity bias for user groups differing in mainstreaminess and gender, (vi) understanding popularity bias mitigation and amplification, and (vii) studying long-term dynamics of fairness in algorithmic decision support. Additionally, this chapter discusses reproducibility aspects of the presented research results and findings.

Chapter 4 - Outlook and Future Research

Chapter 4 gives an outlook into future research directions based on the results, scientific contributions, and findings of this habilitation.

Appendix A and B

The appendices describe the author's own contributions to the publications presented in this habilitation, as well as include the full texts of these publications.

Contents

Acknowledgements	v
Abstract	vi
Kurzfassung (Abstract in German)	vii
1 Introduction	1
1.1 Scientific Positioning of this Habilitation	1
1.2 Main Publications	4
2 Related Work and Background	7
2.1 Main Concepts of Recommender Systems	7
2.2 Transparency and Cognitive Models in Recommender Systems . .	10
2.2.1 The Role of Psychology in Recommender Systems	11
2.2.2 Cognitive-inspired Recommendations	11
2.3 Privacy and Limited Preference Information in Recommender Systems	14
2.3.1 Privacy-aware Recommendations	14
2.3.2 Limited Availability of User Preference Information	16
2.4 Fairness and Popularity Bias in Recommender System	17
2.4.1 Fairness in Algorithmic Decision Support	17
2.4.2 Measuring, Understanding, and Mitigating Popularity Bias	19
3 Scientific Contributions	21
3.1 Transparency and Cognitive Models in Recommender Systems . .	21
3.2 Privacy and Limited Preference Information in Recommender Systems	24
3.3 Fairness and Popularity Bias in Recommender Systems	27
3.4 Summary of Contributions and Reproducibility of Research Results	31
4 Outlook and Future Research	34
Bibliography	36
Appendices	60
A Own Contributions to Main Publications	61

B Full Texts of Main Publications	67
B.1 Transparency and Cognitive Models in Recommender Systems	67
P1 Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach (<i>International Journal of Human-Computer Interaction</i> , 2018)	67
P2 Modeling Popularity and Temporal Drift of Music Genre Preferences (<i>Transactions of the International Society for Music Information Retrieval</i> , 2020)	87
P3 Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations (<i>HUMANIZE @ ACM IUI</i> , 2020)	102
P4 Psychology-informed Recommender Systems (<i>Foundations and Trends in Information Retrieval</i> , 2021)	113
P5 Integrating the ACT-R Framework and Collaborative Filtering for Explainable Sequential Music Recommendation (<i>RecSys</i> , 2023)	179
B.2 Privacy and Limited Preference Information in Recommender Systems	188
P6 Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence (<i>RecSys</i> , 2018)	188
P7 Using Autoencoders for Session-based Job Recommendations (<i>User Modeling and User-Adapted Interaction</i> , 2020)	194
P8 Robustness of Meta Matrix Factorization Against Strict Privacy Constraints (<i>ECIR</i> , 2021)	237
P9 ReuseKNN: Neighborhood Reuse for Differentially-Private KNN-Based Recommendations (<i>ACM Transactions on Intelligent Systems and Technology</i> , 2023)	251
P10 Differential Privacy in Collaborative Filtering Recommender Systems: A Review (<i>Frontiers in Big Data</i> , 2023)	281
B.3 Fairness and Popularity Bias in Recommender Systems	289
P11 The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study (<i>ECIR</i> , 2020)	289
P12 Support the Underground: Characteristics of Beyond-Mainstream Music Listeners (<i>EPJ Data Science</i> , 2021)	298
P13 Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected? (<i>RecSys</i> , 2021)	325
P14 Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems (<i>BIAS @ ECIR</i> , 2022)	332
P15 What Drives Readership? Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations (<i>ECIR</i> , 2022)	344
P16 A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations (<i>BIAS @ ECIR</i> , 2023)	353
P17 Long-Term Dynamics of Fairness: Understanding the Impact of Data-Driven Targeted Help on Job Seekers (<i>Nature Scientific Reports</i> , 2023)	370

Acknowledgements

I would like to thank a number of excellent people who supported me in the last six years, while working on his habilitation. First, Stefanie Lindstaedt, the former head of the Institute of Interactive Systems and Data Science (ISDS) of TU Graz, and former CEO of Know-Center Graz, for supporting me during my whole habilitation process, and in building up my own group, the FAIR-AI research area. I wish you all the best for your next endeavors, namely being the founding president of the Institute of Digital Sciences Austria (IDSA) in Linz. I also thank Frank Kappe, who is the new head of ISDS, and Roman Kern, who is taking over the scientific leadership of Know-Center Graz, for supporting me in my final steps of this habilitation process. Another special thank you goes to Elisabeth Lex, my Ph.D. mentor and former area head of the Social Computing group at Know-Center Graz. Thank you for all the valuable advices with respect to this habilitation, and for the great collaborations on our joint publications.

Within ISDS and Know-Center Graz, I would like to thank my FAIR-AI research group for the great support, and for jointly working together on all our research projects. Special thanks go to Emanuel Lacic, now working at InfoBip in Zagreb, for helping me in building up this research group, to Simone Kopeinik for bringing in all these excellent research ideas with respect to fairness and bias in AI, to Dieter Theiler for providing important software development know-how needed to deploy recommender systems in practice, to Leon Fadljevic for the support in all our data science projects, to Peter Muellner for being an excellent PhD student, and for always finishing the planned tasks perfectly in time, and, to Tomislav Duricic for being another great PhD student, and for always being highly motivated independent of the given task. Finally, I also thank Jana Lasser for providing me with valuable advices for finishing this habilitation.

I have been lucky to collaborate with a lot of brilliant people in the last years: I would like to thank Markus Schedl from Johannes Kepler University in Linz for the fruitful collaborations and research projects we conducted so far, Nava Tintarev from Maastricht University for being a great host during my research visit in 2021, Robin Burke from University Boulder-Colorado for the interesting discussions on fairness aspects of recommender systems, and Eva Zangerle from University Innsbruck and Christine Bauer from Paris-Lodron University Salzburg for the great collaborations. I am looking forward to our discussions at the Dagstuhl Seminar in 2024! I also want to thank the members of the commission and the reviewers of this habilitation for their work and valuable feedback. Finally, I thank my friends and family, and here especially my wife Tea, for their support, understanding, and love. Rest in Peace, Philipp, we will never forget you!

Abstract

Recommender systems have become a pervasive part of our daily online experience by analyzing past usage behavior to suggest potential relevant content, e.g., music, movies, or books. Today, recommender systems are one of the most widely used applications of artificial intelligence and machine learning. Therefore, regulations and requirements for trustworthy artificial intelligence, for example, the European AI Act, which includes notions such as transparency, privacy, and fairness are also highly relevant for the design, development, evaluation, and deployment of recommender systems in practice. This habilitation elaborates on aspects related to these three notions in the light of recommender systems, namely: (i) transparency and cognitive models, (ii) privacy and limited preference information, and (iii) fairness and popularity bias in recommender systems.

Specifically, with respect to aspect (i), we highlight the usefulness of incorporating psychological theories for a transparent design process of recommender systems. We term this type of systems psychology-informed recommender systems. We also use models of human memory theory to develop cognitive-inspired algorithms for tag and music recommendations, and find that these algorithms are capable of outperforming related methods in terms of recommendation accuracy. Additionally, we show that cognitive models can further contribute to transparency aspects of recommender systems by illustrating how the models' components have contributed to generate the recommendation lists.

In aspect (ii), we study and address the trade-off between accuracy and privacy in differentially-private recommendations. We design a novel recommendation approach for collaborative filtering based on an efficient neighborhood reuse concept, which reduces the number of users that need to be protected with differential privacy. Furthermore, we address the related issue of limited availability of user preference information, e.g., click data, in the settings of session-based and cold-start recommendations, by using, e.g., variational autoencoders.

With respect to aspect (iii), we analyze popularity bias in collaborative filtering-based recommender systems. We find that the recommendation frequency of an item is positively correlated with this item's popularity. This also leads to the unfair treatment of users with little interest in popular content, since these users receive worse recommendation accuracy results than users with high interest in popular content. We also find that female users are more strongly affected by the algorithms' amplification of popularity bias. Besides, we present results of an online study on popularity bias mitigation in the field of news article recommendations. Finally, we study long-term fairness dynamics in algorithmic decision support in the labor market using agent-based modeling techniques.

Kurzfassung

Empfehlungssysteme sind zu einem allgegenwärtigen Teil unserer täglichen Online-Erfahrung geworden, indem sie das vergangene Nutzerverhalten analysieren, um relevante Inhalte vorzuschlagen, beispielsweise Musik, Filme oder Bücher. Mittlerweile gehören Empfehlungssysteme zu den am weitesten verbreiteten Anwendungen der künstlichen Intelligenz und des maschinellen Lernens. Daher sind Vorschriften für vertrauenswürdige künstliche Intelligenz, welche Anforderungen wie Transparenz, Datenschutz und Fairness umfassen, für die Entwicklung von Empfehlungssystemen relevant. Diese Habilitation untersucht Empfehlungssysteme in Hinblick auf Aspekte, die mit diesen Anforderungen verknüpft sind, nämlich: (i) Transparenz und kognitive Modelle, (ii) Datenschutz und limitierte Präferenz-Informationen, sowie (iii) Fairness und Popularitätsverzerrungen.

Bezüglich Aspekt (i) zeigen wir den Nutzen von psychologischen Theorien für einen transparenten Designprozess von Empfehlungssystemen. Wir bezeichnen diese als Psychologie-inspirierte Empfehlungssysteme. Zusätzlich verwenden wir Modelle der menschlichen Gedächtnistheorie für die Entwicklung von Empfehlungssystemen und zeigen, dass diese Algorithmen verwandte Methoden, in Bezug auf die Vorhersagegenauigkeit, übertreffen. Darüber hinaus zeigen wir, dass die kognitiven Modelle dazu verwendet werden können, um zu illustrieren, welche Komponenten für die Empfehlungsgenerierung wichtig gewesen sind.

In Hinblick auf Aspekt (ii) untersuchen wir die Beziehung zwischen Genauigkeit und Datenschutz in Empfehlungssystemen, die Differential Privacy verwenden. Wir entwickeln einen neuartigen Empfehlungsalgorithmus, der auf einem effizienten Konzept zur Wiederverwendung von Nachbarschaften im kollaborativen Filtern basiert. Dadurch kann der notwendige Einsatz von Differential Privacy minimiert werden. Darüber hinaus adressieren wir ein damit verwandtes Problem, nämlich das der limitierten Nutzerpräferenz-Informationen, z.B., Klick-Daten, durch die Verwendung von z.B., Variational Autoencodern.

Bezüglich Aspekt (iii) analysieren wir den Einfluss der Popularitätsverzerrung auf die Genauigkeit von Empfehlungssystemen. Wir zeigen, dass Popularität und Empfehlungshäufigkeit positiv korreliert sind, welches auch zur unfairen Behandlung von Nutzern führt, die wenig Interesse an populären Inhalten haben. Diese Nutzer erhalten eine geringere Empfehlungsgenauigkeit als Nutzer, die an populären Inhalten interessiert sind. Darüber hinaus zeigen wir, dass weibliche Benutzer stärker von Popularitätsverzerrungen betroffen sind. Wir präsentieren außerdem Ergebnisse einer Online-Studie zur Minderung des Einflusses von Popularitätsverzerrungen. Abschließend untersuchen wir Langzeiteffekte von Fairness in algorithmischen Entscheidungen mittels agentenbasierter Modellierung.

Chapter 1

Introduction

The present postdoctoral thesis is a cumulative habilitation submitted to Graz University of Technology for the scientific subject *Applied Computer Science*. This habilitation summarizes and discusses scientific publications that have been published between 2018 and 2023, i.e., during the habilitation's author's postdoctoral research. This chapter describes the scientific positioning of this habilitation (Section 1.1), and introduces the 17 main publications that constitute this work (Section 1.2). All publications are peer-reviewed, are already published, and contain a digital object identifier (*DOI*). The publications consist of 7 journal articles, 7 conference proceedings contributions, two workshop post-proceedings book chapters, and one workshop paper. The latter was published via the academic distribution service *arXiv* in accordance with the publishing guidelines of the *Workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory* co-located with *ACM IUI 2020*.

1.1 Scientific Positioning of this Habilitation

This habilitation investigates the research field of recommender systems in general, and aspects of transparency and cognitive models, privacy and limited preference information, and fairness and popularity bias in recommender systems in particular. The research field of recommender systems makes use of multiple aspects of *Applied Computer Science*, including (but not limited to) data science, user modeling, personalization, machine learning, information retrieval, human computer interaction, computational social science, and trustworthy artificial intelligence.

More concretely, recommender systems can be seen as one of the most widely used instantiations of machine learning and artificial intelligence, and accompany us in our daily online experience. Thus, recommender systems have become an integral part of our digital life for supporting humans in finding relevant information in information spaces that are too big or complex for manual filtering (e.g., [47, 125, 225]). Since the early implementations of recommendation algorithms (e.g., [223, 224]), these systems analyze past usage behavior in order to build user models, and to suggest items (e.g., movies), or even people in social networks [76], to individual users or to groups of users (e.g., [184, 185]). To build these user models, different techniques have been employed, including traditional

approaches such as collaborative filtering [80], content-based filtering [176], and hybrid recommendations [45], and more recent approaches based on latent representations (or embeddings) and deep learning [57, 265, 278]. Thus, also different types of data sources are utilized for generating recommendations, e.g., preference information such as ratings, and content features of items (see Section 2.1 for more details on recommender systems in general). Apart from that, recent research has illustrated the multi-stakeholder nature of recommender systems [1, 3]. Thus, not only users are affected by recommendations, but also other stakeholders [120], such as platform operators or item providers (e.g., music artists). Balancing the goals of multiple stakeholders is an active research topic, and further illustrates the far-reaching impact of recommender systems on society [48].

The uptake of recommender systems both in academia and industry [52, 119, 126], as well as their human-centric nature, emphasizes that current regulations and requirements for trustworthy artificial intelligence (AI) are also of high importance for the deployment of recommender systems [69]. Trustworthiness entails multiple notions that have been defined and categorized by the European Commission and institutions in other countries. This has led to different regulations and requirements, for example, the *EU Artificial Intelligence Act* [59], or the *United States Regulatory Development Relating to AI* [259]. Although there are differences between these regulations and requirements, all of them include notions related to transparency, privacy, and fairness in AI. These aspects are also highly relevant under the lens of recommender systems, as indicated by recent related research investigating trustworthy recommender systems [83, 84, 96]. This habilitation contributes to this line of research in the following fields:

Transparency and Cognitive Models in Recommender Systems

One issue of modern recommender systems algorithms based on deep learning techniques (e.g., [265, 278]) is that these approaches are mostly based on principles of artificial intelligence rather than human intelligence. This could lead to non-transparent algorithmic decisions that are hard to understand by the system's users [244]. Apart from methods coming from the fields of explainable AI [193] and explainable recommender systems [255], one way to address this issue is to use theories from psychology to enhance the transparency of recommendation models.

This habilitation uses cognitive models of human memory for a transparent design of recommendation approaches [150, 154, 156, 168, 240]. Specifically, we show that models of human episodic memory and activation processes in human memory can help to create transparent and accurate recommendation models. In this respect, we also illustrate to what extent the components of these models contribute to the generation of the recommendation lists [196]. Finally, we survey and categorize research at the intersection of recommender systems and psychology, which we term *psychology-informed recommender systems* [169].

Privacy and Limited Preference Information in Recommender Systems

Recommender systems need to analyze user preference information to calculate personalized recommendations, which could lead to multiple privacy threats to

users [90]. This includes the inference of users' sensitive information (e.g., gender), or the disclosure of users' preference information (e.g., who bought what) via the analysis of generated recommendation lists by untrusted third parties (e.g., [33, 49, 277]). Thus, privacy has become a key requirement for personalized recommender systems, especially in the light of current data protection initiatives such as the *European General Data Protection Regulation (GDPR)*. Therefore, privacy is related to the issue of limited availability of user preference information (e.g., clicks or ratings) due to the restricted utilization of users' preference information as a result of data protection initiatives [40, 62], and due to the increased privacy concerns of users (e.g., users are not willing to share preference information or to sign in to the system) [137, 164, 189]. This could lead to the user cold-start problem [235] and session-based recommendation settings, since long-term user preferences (including past preferences of the target user) are unavailable [124].

This habilitation investigates issues of limited preference information by addressing the user cold-start problem using recommendations based on users' trust connections [72], and by studying the usefulness of variational autoencoders for session-based job recommendations [162]. Additionally, we study varying privacy constraints of users in a matrix factorization-based recommender system using *meta learning* [197]. We also address the privacy-accuracy trade-off in differentially private recommender systems by utilizing an efficient neighborhood reuse concept [201]. Finally, we survey and categorize the literature on employing *differential privacy* for collaborative filtering recommender systems [200].

Fairness and Popularity Bias in Recommender Systems

Although bias and fairness in algorithmic decision support and machine learning is a research topic that has gained a lot of attraction in recent years [37, 158, 190], the reflection and replication of biases is still an open research problem in the field of interactive systems in general [91, 163], and recommender systems in particular [55, 179, 192]. Here, especially popularity bias is a common issue in recommender systems based on collaborative filtering, and leads to the underrepresentation of unpopular content in personalized recommendation lists [9, 20, 82].

The research presented in this habilitation shows that this popularity bias unfairly affects users with little interest in popular content, since this user group receives lower recommendation accuracy than users interested in popular content [145, 151, 152, 155]. We also find that recommendation algorithms could amplify popularity bias for female users [166], and that content-based recommendations can help to mitigate popularity bias [159]. Additionally, we study long-term fairness in algorithmic decision support in the labor market, and find that there is a trade-off between *individual* and *group fairness* in this setting [237].

Reproducibility Aspects of this Habilitation

The reproducibility of recommender systems research results is of utmost importance to be able to track the scientific progress in the field (e.g., [32, 86]). This habilitation provides code and data resources that should foster the reproducibility of the presented research contributions (see Section 3.4 for a full list).

1.2 Main Publications

Table 1.1 lists the 17 main publications of this habilitation. I have selected 5 publications for each of the first two research topics described beforehand. For the third research topic, fairness and popularity bias in recommender systems, I have selected 7 publications, since this is the research topic I have investigated most recently (here, my first paper was published in 2020). Within these three research fields, the publications are sorted by publication year in ascending order. Overall, each publication is assigned a unique ID, i.e., $[P_i]$, where $i = 1 \dots 17$.

In the first field, transparency and cognitive models in recommender systems, the list of publications contains three studies, in which cognitive models are employed for a transparent design process of recommender systems, i.e., one recommendation approach based on a model of human episodic memory $[P_1]$, and two approaches based on models formalizing activation processes in human memory $[P_2]$ $[P_3]$. Furthermore, it lists a survey on psychology-informed recommender systems $[P_4]$. Another publication illustrates to what extent components of cognitive models contribute to the generation of the recommendation lists $[P_5]$.

The second research field contains two studies on addressing the issue of limited availability of user preference information: one addresses the user cold-start problem using trust-based collaborative filtering $[P_6]$, and one employs variational autoencoders for session-based job recommendations $[P_7]$. Table 1.1 also contains three publications on privacy-aware recommender systems, one addressing varying privacy constraints of users $[P_8]$, one addressing the accuracy-privacy trade-off of differentially private recommender systems $[P_9]$, and one surveying the use of *differential privacy* in collaborative filtering recommender systems $[P_{10}]$.

In the third field, Table 1.1 contains two publications that study popularity bias and characteristics of “niche” users in music recommendations $[P_{11}]$ $[P_{12}]$. One paper further studies if users of different genders are equally affected by popularity bias in music recommendations $[P_{13}]$, and another paper studies popularity bias in multimedia recommendation domains $[P_{14}]$. Furthermore, this list contains an online study on popularity bias mitigation in news article recommendations $[P_{15}]$. Another paper analyzes miscalibration and popularity bias amplification in recommendations $[P_{16}]$. Finally, one journal article studies long-term dynamics of fairness in algorithmic decision support in the labor market $[P_{17}]$.

Table 1.1: List of main publications selected by the author of this habilitation.

No.	Publication
Transparency and Cognitive Models in Recommender Systems	
$[P_1]$	Seitlinger, P., Ley, T., Kowald, D. , Theiler, D., Hasani-Mavriqi, I., Dennerlein, S., Lex, E., Albert, D. (2018). Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach. <i>International Journal of Human-Computer Interaction</i> , 34:6, pp. 557-575. DOI: https://doi.org/10.1080/10447318.2017.1379240

-
- [P2] Lex, E.*, **Kowald, D.***, Schedl, M. (2020). Modeling Popularity and Temporal Drift of Music Genre Preferences. *Transactions of the International Society for Music Information Retrieval*, 3:1, pp. 17-30. (*equal contribution) DOI: <https://doi.org/10.5334/tismir.39>
- [P3] **Kowald, D.***, Lex, E.*, Schedl, M. (2020). Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations. In *4th Workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory (HUMANIZE @ ACM IUI'2020)*. (*equal contribution) DOI: <https://doi.org/10.48550/arXiv.2003.10699>
- [P4] Lex, E., **Kowald, D.**, Seitlinger, P., Tran, T., Felfernig, A., Schedl, M. (2021). Psychology-informed Recommender Systems. *Foundations and Trends in Information Retrieval*, 15:2, pp. 134–242. DOI: <https://doi.org/10.1561/1500000090>
- [P5] Moscati, M., Wallmann, C., Reiter-Haas, M., **Kowald, D.**, Lex, E., Schedl, M. (2023). Integrating the ACT-R Framework and Collaborative Filtering for Explainable Sequential Music Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems (RecSys'2023)*, pp. 840–847. DOI: <https://doi.org/10.1145/3604915.3608838>
-
- Privacy and Limited Preference Information in Recommender Systems**
-
- [P6] Duricic, T., Lacic, E., **Kowald, D.**, Lex, E. (2018). Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys'2018)*, pp. 446–450. DOI: <https://doi.org/10.1145/3240323.3240404>
- [P7] Lacic, E., Reiter-Haas, M., **Kowald, D.**, Dareddy, M., Cho, J., Lex, E. (2020). Using Autoencoders for Session-based Job Recommendations. *User Modeling and User-Adapted Interaction*, 30, pp. 617–658. DOI: <https://doi.org/10.1007/s11257-020-09269-1>
- [P8] Muellner, P., **Kowald, D.**, Lex, E. (2021). Robustness of Meta Matrix Factorization Against Strict Privacy Constraints. In *Proceedings of the 43rd European Conference on Information Retrieval (ECIR'2021)*, pp. 107-119. DOI: https://doi.org/10.1007/978-3-030-72240-1_8
- [P9] Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). ReuseKNN: Neighborhood Reuse for Differentially-Private KNN-Based Recommendations. *ACM Transactions on Intelligent Systems and Technology*, 14:5, pp. 1-29. DOI: <https://doi.org/10.1145/3608481>
- [P10] Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). Differential Privacy in Collaborative Filtering Recommender Systems: A Review. *Frontiers in Big Data*, 6:1249997, pp. 1-7. DOI: <https://doi.org/10.3389/fdata.2023.1249997>

Fairness and Popularity Bias in Recommender Systems

- [P11] **Kowald, D.**, Schedl, M., Lex, E. (2020). The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In *Proceedings of the 42nd European Conference on Information Retrieval (ECIR'2020)*, pp. 35-42. DOI: https://doi.org/10.1007/978-3-030-45442-5_5
- [P12] **Kowald, D.**, Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E. (2021). Support the Underground: Characteristics of Beyond-Mainstream Music Listeners. *EPJ Data Science*, 10:14. DOI: <https://doi.org/10.1140/epjds/s13688-021-00268-9>
- [P13] Lesota, O., Melchiorre, A., Rekabsaz, N., Brandl, S., **Kowald, D.**, Lex, E., Schedl, M. (2021). Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected? In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'2021)*, pp. 601-606. DOI: <https://doi.org/10.1145/3460231.3478843>
- [P14] **Kowald, D.**, Lacic, E. (2022). Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2022)*. Communications in Computer and Information Science, vol. 1610, pp. 1-11. DOI: https://doi.org/10.1007/978-3-031-09316-6_1
- [P15] Lacic, E., Fadljevic, L., Weissenboeck, F., Lindstaedt, S., **Kowald, D.** (2022). What Drives Readership? An Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations. In *Proceedings of the 44th European Conference on Information Retrieval (ECIR'2022)*, pp. 172-179. DOI: https://doi.org/10.1007/978-3-030-99739-7_20
- [P16] **Kowald, D.***, Mayr, G.*., Schedl, M., Lex, E. (2023). A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2023)*. Communications in Computer and Information Science, vol. 1840, pp. 1-16. (*equal contribution) DOI: https://doi.org/10.1007/978-3-031-37249-0_1
- [P17] Scher, S., Kopeinik, S., Truegler, A., **Kowald, D.** (2023). Long-Term Dynamics of Fairness: Understanding the Impact of Data-Driven Targeted Help on Job Seekers. *Nature Scientific Reports*, 13:1727. DOI: <https://doi.org/10.1038/s41598-023-28874-9>
-

In Appendix Chapter A, I comment on my own contributions to these 17 papers. As described in Appendix Chapter A, I have contributed substantially to all 17 publications, and for 10 of these publications I am also either first or last author. The full texts of these publications can be found in Appendix Chapter B. Except for [P4], where I altered the formatting slightly due to copyright restrictions of the journal, I use the published journal and conference formats.

Chapter 2

Related Work and Background

This chapter describes relevant research and background related to the scientific contributions of this habilitation. First, the main concepts of recommender systems are briefly outlined in Section 2.1, followed by relevant background with respect to transparency and cognitive models in recommender systems in Section 2.2. Next, the topic of privacy and limited preference information in recommender systems is briefly discussed in Section 2.3. Finally, Section 2.4 gives a compact overview of fairness and popularity bias in recommender systems. This chapter also summarizes our own research related to these topics, which is then outlined in relation to the main publications of this habilitation in Chapter 3.

2.1 Main Concepts of Recommender Systems

This section gives a compact overview of recommender systems (i) algorithms, (ii) applications, and (iii) evaluation methods relevant to this habilitation.

Recommender Systems Algorithms

In general, there are three main categories of recommendation algorithms [18, 225]: (i) collaborative filtering (CF), (ii) content-based filtering (CBF), and (iii) hybrid approaches. This habilitation focuses on CF, but also investigates CBF.

Typically, a user-based CF recommender system \mathcal{R}^k generates an estimated rating score for a target user u and a target item i by utilizing the ratings $r_{n,i}$ of k other users that have rated i , i.e., the k nearest neighbors $N_{u,i}^k$ [68]. Therefore, this variant of CF is often referred to as *UserKNN*, i.e., user-based k nearest neighbors. Formally, the estimated rating score $\mathcal{R}^k(u, i)$ for u and i is given by:

$$\mathcal{R}^k(u, i) = \frac{\sum_{n \in N_{u,i}^k} sim(u, n) \cdot r_{n,i}}{\sum_{n \in N_{u,i}^k} sim(u, n)} \quad (2.1)$$

where $sim(u, n)$ is the similarity between target user u and neighbor n . For *UserKNN*, the neighborhood $N_{u,i}^k$ used for generating recommendations for u and i comprises the k most similar neighbors. More formally:

$$N_{u,i}^k = \arg \max_{c \in U_i} sim(u, c) \quad (2.2)$$

where U_i are all users that have rated i and sim is the similarity metric (e.g., Cosine or Pearson [35]). There also exist variations of this algorithm suitable for item relevance prediction and for implicit user preferences (e.g., clicks) [122].

It is also possible to calculate similarities between items based on users' preferences of these items. This variant of CF is termed item-based CF (or *ItemKNN*) and has advantages in cases when user profiles change quickly [228]. *ItemKNN* was introduced as the main recommendation algorithm by Amazon.com [174] in 2003. For a comprehensive review of KNN-based CF methods, please see [204], and for a survey on CF with side information, please see [243]. Another possibility to incorporate side and context information (e.g., time or location) is by utilizing context-aware recommender systems, as discussed in these works [14, 17, 19].

Another variant of CF is *matrix factorization* (MF), which follows the idea that a user's preferences can be efficiently represented in low-dimensional space [139, 242]. The items are represented in the same low-dimensional space, which enables to generate recommendations by calculating the dot-product between the user and the item vectors. These vectors are often termed *embeddings*, and can be calculated with techniques such as graph neural networks [250], recurrent neural networks [112], neural CF [108], or autoencoders [274]. As described in Section 2.3, in this habilitation, autoencoders are used to address the issue of limited preference information in session-based recommendations [123, 170]. Furthermore, a neural CF approach [108] is used to study differentially-private recommendations. For a comparison of neural network and KNN-based methods, please see [87].

The next type of algorithms, content-based filtering (CBF) [176], utilizes content features of items (e.g., genres, title) to build item profiles to overcome the item cold-start problem (i.e., items with no user preference information). These item profiles are then matched with user profiles that also consist of content features of the consumed items [63]. For representing content features, techniques such as LDA (Latent Dirichlet Allocation) [43] can be used. CBF could suffer from a lack of novelty and diversity, since typically items are recommended that are similar to the items the user has consumed in the past. To overcome this issue, hybrid recommendation approaches [45, 46, 140, 213] combine CBF and CF to get the benefits of both worlds. There exist several ways to combine recommendation algorithms [127], including (i) monolithic, where collaborative and content information is combined in a single recommendation model, (ii) parallelized, where the results of different algorithms are combined using, e.g., a weighted approach, and (iii) pipelined, where one algorithm uses the results of another algorithm as input.

Recommender Systems Applications

This habilitation focuses on four application areas for recommender systems, namely (i) tag recommendations, (ii) music recommendations, (iii) job recommendations, and (iv) news article recommendations. The following paragraphs briefly describe the particularities of these application areas. Tag recommendation systems aim to support users in finding descriptive tags (or keywords) for annotating Web resources [128, 180] (e.g., music tracks or tweets in Twitter). Previous research of this habilitation's author has shown that a user's choice of tags is affected by activation processes in human memory [147, 148], which can be utilized

for a transparent design of tag recommendation models [153] (see Section 2.2).

Similar to recommendations in other multimedia domains [66] (e.g., movies or television items [182]), music recommender systems help users to navigate large content databases, and to find content that suit their taste [234]. However, in contrast to movies or books, music has some distinguished properties that also affect the design of music recommendation algorithms [233]: (i) music may be consumed repeatedly, while movies or books are typically consumed only once or a few times at maximum, (ii) music recommendations can be addressed on different abstraction levels including tracks, albums, artists and genres, (iii) rating data is relatively rare in the music domain, and thus implicit user preferences (e.g., listening events) are an important information source for recommender systems [70], and (iv) domain knowledge (e.g., musical sophistication) may have a high impact on how recommendations are perceived by the music listeners [130].

Next, job recommender systems address a particular recommendation problem, in which open job positions should be matched with job candidates [10, 11]. This differs to other recommendation application domains, since typically every open job position (i.e., the item) can be assigned to only one job candidate (i.e., the user), and vice versa [133]. Additionally, job portals (especially those that offer jobs to students and young talents) often provide the possibility to browse jobs anonymously [160, 221], which then turns the job recommendation problem into a session-based recommendation problem [123, 217]. Limited preference information and anonymous user sessions are also issues of news article recommender systems [64, 116, 194]. Via providing recommendations of currently relevant news articles that match session information (e.g., clicks) of the user, news portals aim to increase user engagement, and to turn anonymous readers into paying subscribers [5]. Finally, another particularity of news recommendations is the short lifetime of items, since many articles are only relevant for one day [209].

Recommender Systems Evaluation Methods

This habilitation considers both online and offline evaluation procedures of recommender systems. Both methods aim to compare the performance of two or more recommendation algorithms, but while online evaluation is performed in a live system, e.g., using A/B tests [94], offline evaluation is performed using collected preferences, typically in the form of training and test sets [53]. Another difference lies in the time when the user preference information is collected: whereas online evaluation collects user preferences after the recommendations are shown to the users, offline evaluation gathers user preferences (i.e., the ground truth data in the test set) before the recommendations are calculated [52, 273]. Online evaluation procedures then measure the actual performance using impact- or value-oriented measurements such as *Click-Through-Rate (CTR)* [121]. In contrast, offline evaluation procedures make use of relevance or performance metrics, which are often borrowed from the information retrieval research field [28, 227].

With respect to offline evaluation metrics, this habitation investigates both accuracy and beyond-accuracy metrics. To measure accuracy [61], error-based metrics for rating prediction such as the *Mean Absolute Error (MAE)* [269], and metrics for ranking quality such as *Precision (P)*, *Recall (R)*, *F1-score (F1)*, *Mean*

Reciprocal Rank (MRR), and *Normalized Discounted Cumulative Gain (nDCG)*, have been proposed in the literature (e.g., [111]).

After decades of accuracy-driven recommender systems evaluation procedures, the research community has argued that being accurate is not the only important objective for a recommender system, and has proposed a set of beyond-accuracy metrics [16, 95, 187]. Here, especially, the concepts of *novelty* and *diversity* are important [51]. Novelty describes the difference between the recommended items and a specific context, which could be the target user’s item history or all users’ item histories in the system [262]. The former, which is also referred to as personalized or user-based novelty, or unexpectedness [13], describes how different the recommendation list is from the items the target user has consumed in the past (i.e., the user item history). This concept is also related to serendipity, which, in addition, takes the relevance of the recommended items into account [56]. The latter, which is also referred to as long-tail novelty or system-based novelty [262], measures the rarity or inverse popularity of the recommended items [54]. This concept is also related to evaluating fairness and popularity bias of recommendations, which is described in more detail in Section 2.4.

All methods and metrics discussed so far solely evaluate the recommender system from a user’s, or consumer’s, perspective. However, in recent years, the multi-stakeholder nature of recommender systems has been highlighted, which not only takes the users, but also the item providers (and maybe even other stakeholders like the system operators) into account [1, 3]. Here, especially integrating and evaluating item provider constraints is becoming an important research topic [251], and is also related to multi-sided fairness aspects of recommender systems [48, 246]. Finally, the reproducibility of recommender systems evaluation procedures is another important and timely topic [60]. Here, the adequate documentation and sharing of source-code and dataset samples used in the evaluation process is a key aspect of reproducibility [32]. Please see Section 3.4, for a discussion of reproducibility aspects related to the contributions of this habilitation.

Summary of own research (1): This habilitation studies a wide range of recommendation algorithms and applications such as tag, music, job, and news article recommendations. Additionally, we investigate both accuracy and beyond-accuracy metrics, and both offline and online evaluation settings. Finally, we discuss reproducibility aspects of the scientific contributions of this habilitation, and provide code and data resources to foster reproducibility.

2.2 Transparency and Cognitive Models in Recommender Systems

This habilitation investigates transparency aspects of recommender systems by following principles of psychology and human cognition for a transparent design process of recommendation algorithms. Another possibility to enhance transparency in recommender systems is by providing explanations for recommendations, which is not investigated in this habilitation. For the field of explainability in recommender systems, please see [207, 253, 254, 255].

2.2.1 The Role of Psychology in Recommender Systems

Already, early research in the field of recommender systems was influenced by the fact that humans' decision-making processes are impacted by their social surroundings, which also motivated the implementation of the first collaborative filtering-based recommendation algorithms [223, 224]. In order to create more human-centric recommendations, additional psychological characteristics of users were incorporated in the design and implementation process of recommender systems [27, 257]. For example, insights from decision psychology [165] were used to study serial position and anchoring effects in recommendations [15, 85, 248], and to show that users are more likely to remember items at the beginning (i.e., *primacy* effect) and the end (i.e., *recency* effect) of a list [249]. Related research also investigated how to incorporate aspects such as personality [256], and affect, e.g., emotion [99] or satisfaction [183, 186], into the recommendation process.

Based on these lines of research, we survey and categorize related work at the intersection of psychology and recommender systems. We term this type of recommender systems *psychology-informed recommender systems* [169], and we identify three main areas: (i) cognitive (or cognition)-inspired, (ii) personality-aware, and (iii) affect-aware recommender systems. Additionally, we connect these areas to aspects of human decision-making, and to aspects of human-centric evaluation design of recommender systems. This habilitation focuses on the first area, namely cognitive-inspired recommendations based on human memory theory, which is described in more detail in the following section.

Summary of own research (2): We highlight the usefulness of incorporating the underlying psychological constructs and theories into a transparent design process of recommender systems. We term this type of recommender system *psychology-informed recommender system*, and categorize it into three types.

2.2.2 Cognitive-inspired Recommendations

This habilitation investigates two cognitive-inspired recommendation approaches: one based on human episodic memory, and another one based on activation processes in human memory. Other types of cognition-aware recommendation approaches, such as stereotype-based recommendations [226], categorization-based recommendations [239], or attention-based user models [238], are discussed in [169].

Recommendations based on Human Episodic Memory

Human episodic memory is the memory of personally experienced events that occurred in a specific context (e.g., a particular day or place, or a given categorization) [258]. The contextual information is essential for retrieving these events. MINERVA2 [113] is a model that accounts for episodic memory-based human behavior such as categorization [114], and recognition [115]. MINERVA2 distinguishes between a long-term or secondary memory that holds the episodic memory traces (i.e., the events along with the context information), and a working or primary memory that communicates with the secondary memory by sending retrieval cues (e.g., current context information), and receiving matching events.

In our own research [156, 240], we employ MINERVA2 to implement a tag recommendation algorithm called *Search of Memory (SoMe)*. *SoMe* mimics a user’s search of memory when assigning tags to bookmark a Web resource. Therefore, we encode episodic memory traces using the categories assigned to previously bookmarked Web resources of this user. Specifically, *SoMe* implements MINERVA2’s distinction between the primary and secondary memory in a way that the primary memory represents the Web resource to be tagged in terms of the resource’s categories, and to search the secondary memory for tags that are assigned to Web resources with similar categories. These tags are then recommended to the user. Via user studies, we find that *SoMe* provides higher tag recommendation acceptance than a popularity-based baseline approach [156, 240] (see Section 3.1).

Recommendations based on Activation Processes in Human Memory

Human memory is very efficient in making memory units quickly available when they are needed [39, 211]. More formally, human memory tunes the activation of its units to statistical regularities of the current context and environment [24]. These so-called activation processes in human memory are formalized in the cognitive architecture ACT-R [23]. ACT-R is short for “Adaptive Control of Thought – Rational”, and differs between two long-term memory modules: (i) declarative memory, which holds factual knowledge (i.e., what something is), and (ii) procedural memory, which consists of action sequences (i.e., how to do something) [22].

This habilitation focuses on the declarative memory module, which contains the *activation equation* of human memory. The *activation equation* determines the usefulness, i.e., the activation level A_i , of a memory unit i (e.g., a specific item or item category the user has interacted with in the past) for a user u in the current context. It combines a *base-level* activation with an *associative* activation, which depends on the weight W_j , and the strength of association $S_{j,i}$ [22]:

$$A_i = B_i + \sum_j W_j \cdot S_{j,i} \quad (2.3)$$

where B_i represents the *base-level* activation of i , which quantifies its general usefulness by considering how frequently and recently it has been used in the past. It is defined by the *base-level learning (BLL) equation* [24]:

$$B_i = \ln \left(\sum_{j=1}^n t_j^{-d} \right) \quad (2.4)$$

where n is the frequency of i ’s occurrences in the past (i.e., how often u has interacted with i), and t_j is the time since the j^{th} occurrence of i (i.e., the recency of i). The exponent d accounts for the time-dependent decay of item exposure, which means that each unit’s activation level decreases in time according to a power function. The second part of Equation 2.3 represents the *associative activation* that tunes B_i to the current context. The current context can be defined by any contextual element j that is relevant to the current situation, and via learned associations, the contextual elements can increase i ’s activation.

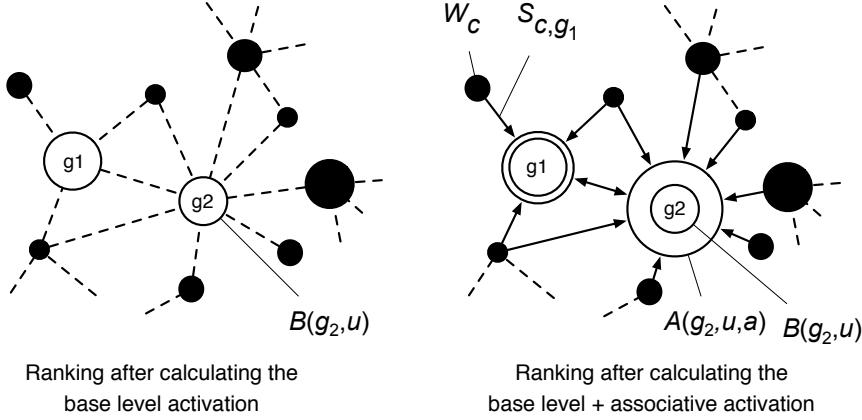


Figure 2.1: An example illustrating the difference between the *BLL equation* (left panel) and the *activation equation* (right panel). Here, unfilled nodes represent target genres g_1 and g_2 , and black nodes represent contextual genres. For g_1 and g_2 , the node sizes represent the activation levels, and for the contextual genres, the node sizes represent the weights W_c . The association strength $S_{c,g}$ is represented by the length of each edge. We see a different ranking of the genres in the two settings, which illustrates the importance of the associative activation [150].

Figure 2.1 illustrates the difference between the base-level activation and the associative activation in the case of a music recommendation system that aims to rank relevant music genres for a given user. The left panel shows the ranking of two genres g_1 and g_2 according to the *BLL equation*. Here, g_1 would have a higher activation level than g_2 based on past usage frequency and recency. The right panel shows the ranking of these genres according to the *activation equation*, which also takes associations with contextual genres into account (e.g., music genres that are relevant in the current situation). Using the combined base-level and associate activation, the ranking changes, and g_2 would have a higher activation level than g_1 [150]. The declarative memory module also contains some additional components. One example is the valuation component [131], which determines the *value* attributed by u to i (e.g., interaction time or frequency [220]).

In our research, we use the *BLL equation* and *activation equation* for a transparent design, implementation, and evaluation process of two music recommendation algorithms [150, 168]. We show that these cognitive-inspired approaches outperform related baselines in terms of recommendation accuracy. Additionally, we illustrate to what extent the components of ACT-R contribute to the generation of the music recommendation lists [196]. In a recently accepted paper [154], we discuss transparency aspects of additional components of ACT-R.

Summary of own research (3): We use models of human episodic memory (i.e., MINERVA2), and activation process in human memory (i.e., ACT-R) for a transparent design, implementation, and evaluation process of recommendation algorithms. We also illustrate to what extent the components of ACT-R (e.g., *BLL*) contribute to the generation of the recommendation lists.

2.3 Privacy and Limited Preference Information in Recommender Systems

This section gives an overview of privacy-aware recommendations. Since the users' privacy concerns could also lead to the limited availability of preference information (e.g., users disclose their preferences, or do not sign in to the system), this section also gives an overview of session-based and cold-start recommendations.

2.3.1 Privacy-aware Recommendations

In terms of privacy, this habilitation focuses on differentially-private recommendations. This section also briefly discusses privacy aspects of recommender systems.

Privacy Aspects of Recommender Systems

Recommender systems need to store and process user preference information, which could lead to potential privacy risks to its users [90]. This includes the inference of private information. Here, related research has shown that inference attacks can be used to derive a user's sensitive information (e.g., gender [245]) based on the information shared with the recommender system [33, 129, 268]. For example, in k -nearest neighbor-based recommender systems, the use of neighbors' preference information in the recommendation process can pose a privacy risk to the neighbors [218, 276]. In this way, the preference information of the neighbors can be uncovered, or the neighbors' identities (or sensitive attributes) can be revealed. Other inference attacks in recommender systems work by generating fake users, i.e., sybils, based on the limited knowledge of a victim's preferences. These sybils isolate the victim utilized as a neighbor, and compromise its privacy [49].

Different privacy-preserving technologies have been used to mitigate the users' privacy risks, including *homomorphic encryption*, *federated learning*, and *differential privacy*. While *homomorphic encryption* techniques aim to generate privacy-aware recommendations by employing encrypted user preference information [275], *federated learning* techniques build on the assumption that sensitive user information should never leave the user's device [25, 272]. Finally, *differential privacy* protects the users by introducing noise into the recommendation process [73].

Our own research focuses on using *differential privacy*. Additionally, we study how limiting the preference information of users can help to increase privacy. Therefore, we use the concept of *meta learning* [173] to calculate recommendations based on a minimal amount of user preference information. With this, we study privacy constraints of users (e.g., willingness to share preference information) [197]. We find that users with small profiles can afford a higher degree of privacy than users with large profiles, and that *meta learning* is helpful for increasing the robustness against the users' privacy constraints (see Section 3.2).

Differentially-Private Recommendations

The aim of differentially-private recommendations is to inject randomness and noise into the recommendation calculation process to mitigate the inference risk

of users' preference information [89, 188]. This habilitation focuses on a specific attack, which can be addressed by using *differential privacy*. Here, a user with malicious intent, i.e., the *adversary* a , tries to infer preference information (here, rating scores) of a specific neighbor n in user-based k -nearest neighbor CF (i.e., *UserKNN*) [49]. In this attack scenario, the adversary a has some prior knowledge about n , such as publicly available rating information P of n that could have been inferred from, e.g., product reviews. Using P , a modifies its own user profile R_a such that it (partially) matches n 's profile, which increases the likelihood of n being used as a neighbor for calculating a 's recommendations. With this, a queries estimated rating scores from the recommender system, i.e., $\mathcal{R}^k(a) = \{\mathcal{R}^k(a, i_1), \mathcal{R}^k(a, i_2), \dots, \mathcal{R}^k(a, i_l)\}$, where $\mathcal{R}^k(a, i_j)$ is the estimated rating score for item $i_j \in Q_a$, and Q_a is the set of a 's rating queries. Then a aims to infer rating information r_{n,i_j} of a neighbor n for item i_j used to generate the estimated rating scores. More formally, this is given by:

$$Pr[r_{n,i_1}, r_{n,i_2}, \dots, r_{n,i_l} | \mathcal{R}^k(a, i_1), \mathcal{R}^k(a, i_2), \dots, \mathcal{R}^k(a, i_l), P \cup R_a] \quad (2.5)$$

To mitigate the inference risk of n 's rating information, different variants of *differential privacy* such as the *Laplace input perturbation* [75] or *plausible deniability* [41] can be used. This habilitation utilizes *randomized responses* [267] to establish *plausible deniability*. Specifically, a privacy mechanism m_{DP} is applied to the neighbors' ratings to generate the differentially-private set of ratings \tilde{R} :

$$\tilde{R} = \{m_{DP}(r_{n,i}) : n \in N_{u,i}^k\} \quad (2.6)$$

Via *randomized responses*, neighbors can plausibly deny that their real rating was used in the recommendation process. In detail, the privacy mechanism m_{DP} flips a fair coin, and if the coin is heads, the neighbor's real rating is used in the recommendation calculation. If the coin is tails, m_{DP} flips a second fair coin to decide whether the neighbor's real rating, or a random rating drawn from a uniform distribution over the range of ratings, is used. With this, the adversary a does not know if the utilized rating is real, or random, which leads to the guarantees of *differential privacy* [75]. However, the randomness introduced to the users' preference information typically leads to accuracy drops, and thus also to a fundamental trade-off between accuracy and privacy [38].

In our research, we address this accuracy-privacy trade-off by proposing a novel differentially-private recommendation approach termed *ReuseKNN* [201]. *ReuseKNN* aims to reduce the number of users that need to be protected via *differential privacy* by employing an efficient neighborhood reuse concept. With this, the majority of users (we call them *secure* users) are rarely used in the recommendation process and thus, do not need protection, while some highly reusable users (we call them *vulnerable* users) can be protected with *differential privacy*. Figure 2.2 schematically illustrates our approach, and shows that the fraction of *secure* users is substantially larger in the case of *ReuseKNN* compared to traditional *UserKNN*. We also find that this leads to higher recommendation accuracy compared to a fully differentially-private recommender system (see Section 3.2). Additionally, we survey, analyze, and categorize the use of *differential privacy* in 26 papers published in recommender systems-relevant venues [200].

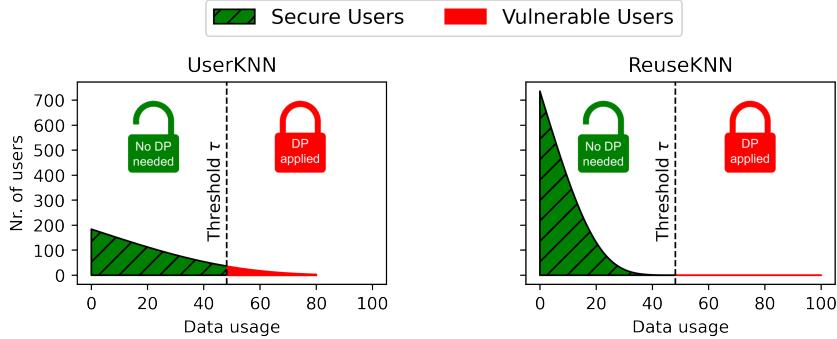


Figure 2.2: Schematic illustration of the data usage (i.e., how often a user is used as a neighbor) distribution of *UserKNN* and *ReuseKNN*. *ReuseKNN* increases the number of *secure* users (green, no *differential privacy* needed) and decreases the number of *vulnerable* users (red, *differential privacy* needs to be applied) compared to *UserKNN*. The dashed line illustrates the data usage threshold τ , a parameter to adjust the maximum data usage for users to be treated as secure.

Summary of own research (4): We study varying privacy constraints of users, e.g., the willingness to share preferences with the recommender system. Additionally, we address the privacy-accuracy trade-off in differentially-private recommendations by employing a neighborhood reuse concept, and survey and categorize the literature on using *differential privacy* in collaborative filtering.

2.3.2 Limited Availability of User Preference Information

Data protection initiatives as well as the users' privacy concerns in recommender systems can lead to the limited availability of preference information [40, 137, 164, 189, 260]. This habilitation investigates this issue in session-based and cold-start recommendation settings, which are discussed in this section.

Session-based Recommendations

Session-based recommender system aim to provide meaningful recommendations in cases where long-term user preferences, or user histories, are not available (e.g., due to users' privacy concerns, or when users do not sign in to the system). The input of a session-based recommender system consists of a typically short item sequence that is observed in the current user session [124, 177]. Different algorithms for session-based recommendations have been proposed, including methods based on k -nearest neighbors [123] or recurrent neural networks [112]. Session-based recommender systems are related to sequence-aware and sequential recommender systems [132], which are not covered in this habilitation. Please see [216] for a detailed overview of sequence-aware and sequential recommendations.

In our research, we employ autoencoders, a specific type of neural network for reducing the dimensionality of data [157], to infer latent session representations, and to generate session-based job recommendations. Specifically, we find that variational autoencoders provide the best results across a set of accuracy and beyond-accuracy evaluation metrics (e.g., system and session-based novelty) [162].

The User Cold-Start Problem

The user cold-start problem in recommender systems refers to users that have interacted with only a few or even with no items at all, i.e., users with limited availability of preference information [235]. Related research has proposed different methods to address the user cold-start problem, including simple popularity-based and unpersonalized approaches [235], location-aware recommendations [161], and trust-based recommendations [77, 181]. This habilitation focuses on trust-based recommendations, which exploit trust statements between users (e.g., user A trusts user B) to create trust networks, and to calculate CF-based recommendations using the connections in these trust networks [103, 104, 208].

In our research, we employ network measures such as *regular equivalence* [109] to calculate trust-based recommendations for cold-start users. Via *regular equivalence*, we do not only find neighbors that share the same trust connections, but also neighbors that have similar structural roles in the trust network (e.g., users that are only connected to influential nodes in the network). We find that our approach outperforms related methods based on, e.g., Jaccard similarity [72].

Summary of own research (5): We address the issue of limited availability of user preference information (e.g., due to users' privacy constraints) in session-based and cold-start recommendation settings. We demonstrate the usefulness of variational autoencoders for session-based job recommender systems. Furthermore, we address the user cold-start problem by employing trust-based recommendations using network measures such as *regular equivalence*.

2.4 Fairness and Popularity Bias in Recommender System

This section gives a brief overview of fairness in algorithmic decision support, and outlines research on popularity bias in recommender systems. For more detailed reviews on fairness-aware recommender systems, please see [65, 78, 266, 271].

2.4.1 Fairness in Algorithmic Decision Support

Fairness in algorithmic decision support and in machine learning applications has gained a lot of attention in recent years, and has been studied especially for binary classification problems [37, 158, 190]. In this problem setting, Y denotes the real outcome to be predicted by the classifier (e.g., the class label, for example if a job applicant has been put into a high- or low-prospect group), and A is the set of protected attributes of an individual, thus the attributes that one must not discriminate against (e.g., gender or race). Furthermore, X denotes non-protected attributes of an individual, and \hat{Y} is the predictor of Y (e.g., to predict to which class the individual belongs), which could depend on X and A . Different definitions of fairness were proposed for such a setting in the literature.

For example, *fairness through unawareness* is satisfied if the predictor \hat{Y} only depends on X and not on A to predict Y , i.e., $\hat{Y} : X \rightarrow Y$. Although this fairness

definition seems to be compelling and simple to implement, it was shown that it is not sufficient in the area of algorithmic decision support since elements of X may contain hidden discriminatory information of A (e.g., race may correlate with the place of residence) [105]. Another definition is based on *individual fairness* [74]. Given that we have a distance metric $d(i, j)$, if two individuals i (with X_i and A_i) and j (with X_j and A_j) are similar according to this metric (so $d(i, j)$ is small), then also their predicted outcomes should be similar: $\hat{Y}(X_i, A_i) \approx \hat{Y}(X_j, A_j)$. One drawback of *individual fairness* is that the definition of $d(i, j)$ requires detailed information of the individuals as well as detailed domain knowledge.

Apart from that, the literature has also provided different definitions for *group fairness*. According to the *statistical parity* (or demographic parity) definition [30], fairness is given if the positive outcome proportion of the predictor $P(\hat{Y} = 1)$ is equal for all A , which, in the binary case with $A \in \{0, 1\}$, is given by:

$$P(\hat{Y} = 1|A = 0) = P(\hat{Y} = 1|A = 1) \quad (2.7)$$

Legally, this metric is often related to the 4/5th rule [100]. This means that the positive outcome ratio between the protected group (i.e., $A = 0$) and the privileged group (i.e., $A = 1$) should be at least 0.8. For example, if the privileged group has a positive outcome proportion of 50%, then the protected group should have a positive outcome proportion of at least 40%. The downside of this metric is that it does not depend on the real outcome Y (only on the predictions \hat{Y}).

In contrast, *equality of opportunity* [106] also takes the real outcome Y into account. The idea is that individuals of the privileged and individuals of the protected group should have equal chance of getting a positive outcome, assuming that the individuals of the groups are qualified for this positive outcome. This can be measured via the true positive rate, which is given by:

$$P(\hat{Y} = 1|Y = 1, A = 0) = P(\hat{Y} = 1|Y = 1, A = 1) \quad (2.8)$$

Equality of opportunity can also be defined using the false negative rate [237]. Additionally, *equalized odds* is a stricter variant of *equality of opportunity* that requires that both the true positive rate and the false positive rate are equal [263]. Research has also found a trade-off between *individual* and *group fairness* [42].

In our research, we employ some of these definitions and adjust them to study long-term dynamics of fairness in algorithmic decision support. Therefore, we develop an agent-based model and evaluate it in a labor market setting [237]. We find that there is a trade-off between different long-term fairness goals, which validates the aforementioned *individual* and *group fairness* trade-off (see Section 3.3). Although, this work does not directly study recommender systems, it sheds light on the usefulness of agent-based modeling for studying algorithmic fairness in the long-term, which is also relevant for the research field of recommender systems.

Summary of own research (6): We study long-term fairness dynamics in algorithmic decision support in a labor market setting using agent-based modeling techniques. We highlight the trade-off between different long-term fairness goals in such a setting (i.e., *individual* and *group fairness*).

2.4.2 Measuring, Understanding, and Mitigating Popularity Bias

In this section, metrics to measure and understand popularity bias, and methods to mitigate popularity bias in recommender systems are briefly discussed.

Popularity Bias Metrics

Research has shown that recommendation algorithms (especially those based on CF) are biased towards popularity, which leads to the overrepresentation of popular items in the recommendation lists [81, 82]. This also leads to the underrepresentation of unpopular items (long-tail items) in the recommendation lists [44, 212]. The literature has proposed different metrics to measure and understand popularity bias from the item and user perspective [20, 136]. This habilitation focuses on three specific ways to measure inconsistencies between user groups with respect to popularity bias: (i) accuracy differences between user groups, (ii) *miscalibration*, and (iii) *popularity lift*. While the first one simply requires comparing the average recommendation accuracy between the groups, *miscalibration* and *popularity lift* are more complex to calculate. Additionally, via *skewness* and *kurtosis*, we measure the asymmetry and “tailedness” of the popularity distributions [34].

In general, *calibration* quantifies the similarity of a genre spectrum between a user profile p and a list of recommendations q [247]. For example, if a user has consumed 80% of rock music and 20% of pop music in the past, then a *calibrated* recommendation list should also contain this genre distribution. Although this metric is not a popularity bias metric by definition, it is often used to measure and understand popularity bias in recommendations [6, 8]. The definition of *calibration* can be reinterpreted in the form of *miscalibration*, i.e., the deviation between p and q [172]. This deviation is calculated using the *Kullback-Leibler (KL)* divergence between the distribution of genres in p , i.e., $p(c|u)$, and the distribution of genres in q , i.e., $q(c|u)$. More formally, for user u , this is given by:

$$KL(p||q) = \sum_{c \in C} p(c|u) \log \frac{p(c|u)}{q(c|u)} \quad (2.9)$$

Here, C is the set of all genres in a given dataset. Therefore, $KL(p||q) = 0$ means perfect *calibration*, and higher $KL(p||q)$ values (i.e., close to 1) mean *miscalibrated* recommendations. The $KL(p||q)$ values can be averaged for a given group g .

In contrast, *popularity lift* measures to what extent recommendation algorithms amplify the popularity bias inherent in the user profiles [7, 8]. Thus, this metric quantifies the disproportionate recommendation of popular items for a given user group g . *Popularity lift* is based on the group average popularity $GAP_p(g)$, which is defined as the average popularity of the items in the user profiles p of group g . Similarly, $GAP_q(g)$ is the average popularity of the recommended items for all users of the group g . Taken together:

$$PL(g) = \frac{GAP_q(g) - GAP_p(g)}{GAP_p(g)} \quad (2.10)$$

$PL(g) > 0$ means that g 's recommendations are too popular, $PL(g) < 0$ means that g 's recommendations are too unpopular, and $PL(g) = 0$ is the ideal value.

In our research, we use these metrics to study popularity bias in recommender systems [145, 151, 155]. We find that “niche” users interested in unpopular content receive worse recommendation quality than users interested in popular content. We study the characteristics of these “niche” users in the field of music recommendations, and identify subgroups that also differ in the recommendation quality they receive [152]. Finally, we also find that music recommendation algorithms could intensify the popularity bias for the group of female users [166].

Popularity Bias Mitigation

Research has proposed different methods to mitigate bias in algorithms, including pre-, in-, and post-processing methods [206]. In the field of recommender systems, especially in-processing and post-processing techniques are used to mitigate popularity bias. Here, in-processing algorithms aim to adjust the recommendation calculation procedure, and to correct the popularity bias using, e.g., *calibration-based* techniques [9, 135]. In contrast, post-processing techniques do not change the recommendation algorithm itself, but the generated recommendation list by using, e.g., re-ranking techniques [4, 26]. Typically, in-processing techniques are the most complex ones to implement, since the underlying algorithm needs to be adapted. However, they are efficient with respect to computational costs. In contrast, one drawback of post-processing techniques is the computational inefficiency of these methods due to the high computational complexity of item re-ranking. However, they can be applied to any given item ranking independent of the underlying algorithm [55]. Finally, the use of content-based recommendation algorithms [63, 176] is another possibility to address popularity bias in recommender systems due to their independence of user preference information [2, 198].

In our research, we study popularity bias mitigation in news article recommender systems for both subscribed users and anonymous session users utilizing content-based recommendations [159]. In an online study that we have conducted together with the Austrian news platform *DiePresse*, we find that personalized and content-based recommendations lead to a more balanced news article readership distribution compared to purely popularity-based recommendations. Thus, we find that readers are not only interested in the most popular and recent news articles, but also in long-tail articles if they match the user preference history, or the preferences tracked in the current session (see Section 3.3).

Summary of own research (7): We analyze popularity bias in collaborative filtering-based recommender systems, and find that “niche” users interested in unpopular content receive worse recommendation accuracy than users interested in popular content. Thus, this “niche” user group is treated in an unfair way by collaborative filtering-based recommender systems. Furthermore, we analyze the characteristics of these users, and study popularity bias mitigation in news article recommender systems using content-based recommendations.

Please note that the aim of this “Related Work and Background” chapter has not been to give a comprehensive review of the various research fields mentioned, but rather to discuss the research and background related to the scientific contributions and publications of this habilitation described in the next chapter.

Chapter 3

Scientific Contributions

This chapter describes the scientific contributions of this habilitation according to the three research topics that are investigated: (i) transparency and cognitive models (Section 3.1), (ii) privacy and limited preference information (Section 3.2), and (iii) fairness and popularity bias (Section 3.3) in recommender systems. Therefore, the 17 publications listed in Table 1.1 are categorized into 7 scientific contributions. For research topics (i) and (ii), this leads to two contributions each, and for research topic (iii), this leads to three contributions, since this topic also covers the most publications of this habilitation. The full texts of these 17 publications can be found in Appendix Chapter B. Additionally, Section 3.4 summarizes the scientific contributions, and elaborates on reproducibility aspects.

3.1 Transparency and Cognitive Models in Recommender Systems

This section summarizes our research on transparency aspects of recommendations by using cognitive modeling techniques. It contains three studies employing cognitive models for a transparent design process of tag and music recommendation algorithms [P1] [P2] [P3], and one survey and categorization of psychology-informed recommender systems [P4] (*Contribution 1*). Additionally, one study illustrates to what extent the components of the cognitive model ACT-R contribute to the generation of music recommendation lists [P5] (*Contribution 2*).

Contribution 1: Using Cognitive Models for a Transparent Design and Implementation Process of Recommender Systems (2018-2021)

[P1] introduces a tag recommendation algorithm termed *SoMe* (*Search of Memory*) based on *MINERVA2* [113], which is a model of human episodic memory (see Section 2.2.2). We implement *SoMe* using our *TagRec* framework [144, 146], and evaluate it in an online study with 18 participants. During the four-weeks study, the participants had to investigate a specific topic (i.e., “designing workplaces that inspire people”) by collecting and tagging three topic-related Web resources per week. For this, the participants were supported with a social bookmarking user interface (based on the *KnowBrain* tool [67]) that contained support via tag rec-

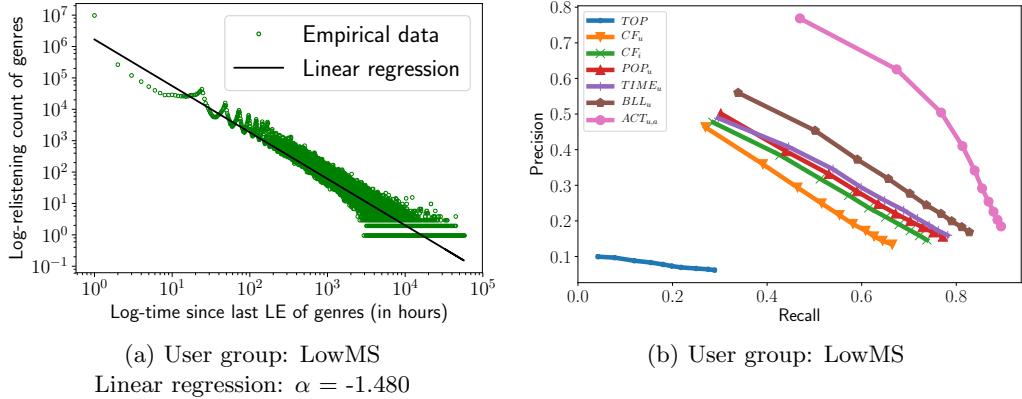


Figure 3.1: (a) Calculation of the BLL equation’s d parameter. On a log-log scale, we plot the relistening count of the genres over the time since their last listening event (LE), and set d to the slopes α of the linear regression lines [168]. (b) Recall/precision plots for $k = 1 \dots 10$ predicted genres of the baselines, and our BLL_u and $ACT_{u,a}$ approaches. $ACT_{u,a}$ achieves the highest accuracy [150].

ommendations. Here, the participants randomly received tag recommendations calculated via *SoMe* or via a conventional *MostPopular* tag recommendation algorithm. Additionally, the participants were divided into two groups at random: (i) *individual*, where the participants only saw their own resources and tags, and (ii) *collaborative*, where the participants also saw the resources and tags of the other users in the group. Thus, in the *collaborative* setting, the tag recommendations were calculated based on the categorized resources and tags of the other users as well. The outcomes of our online study show that, in the *collaborative* setting, *SoMe* provides significantly higher tag recommendation acceptance rates than the *MostPopular* approach. In the *individual* setting, we do not observe a significant difference between the two approaches in terms of recommendation acceptance. Therefore, we find that a cognitive-inspired tag recommendation algorithm based on a transparent model of human episodic memory supports users in *collaborative* tagging settings. We have validated these findings in a follow-up paper using a similar tag recommendation approach termed *3Layers*, which we have presented at the *International World Wide Web conference 2018 (TheWebConf)* [156].

P2 and P3 present the second set of our cognitive-inspired recommendation algorithms based on activation processes in human memory as defined by the cognitive architecture ACT-R [22] (see Section 2.2.2). We introduce two algorithms for a transparent modeling and prediction approach for music genre preferences of users: (i) BLL_u , which implements the *base-level learning* (BLL) equation of ACT-R as described in P2, and (ii) $ACT_{u,a}$, which extends BLL_u , and implements the full *activation equation* of ACT-R as described in P3. We evaluate these approaches using dataset samples containing preferences (listening events) of users of the Last.fm music platform, based on the *LFM-1b* dataset [229, 232].

Figure 3.1 (a) illustrates the impact of time on the re-listening behavior of users in our Last.fm dataset sample. We find that users tend to listen to music

genres to which they have listened to very recently, and that this temporal decay follows a power-law distribution as suggested by the *BLL* equation of ACT-R [24]. We use the slope α of the linear regression of this data to set *BLL*'s d parameter. Figure 3.1 (b) shows the accuracy of our approaches compared to five baseline algorithms: *TOP* suggests the most popular genres in the system, CF_u and CF_i represent user-based and item-based CF, and POP_u and $TIME_u$ suggest the most popular and most recent genres listened to by u , respectively. We find that BLL_u outperforms all baselines, and that $ACT_{u,a}$ outperforms BLL_u by also taking into account the current context of music listening (i.e., the genres of the artist a to which the user u listened to most recently) via the spreading activation component. Our findings show the usefulness of activation processes in human memory for a transparent design process of music recommendation algorithms, which also leads to high recommendation accuracy. We have validated these findings for the task of hashtag recommendations [153, 167], and for the task of music artist recommendations, which we have presented at the *International Society for Music Information Retrieval (ISMIR)* conference 2019 [149].

Finally, P4 surveys and categorizes recommender systems that draw on psychological theories for a transparent design, implementation, or evaluation process of recommendations. We term this type of recommender systems *psychology-informed recommender systems* and categorize them into three groups: (i) cognition (or cognitive)-inspired, (ii) personality-aware, and (iii) affect-aware recommender systems (see Section 2.2.1). We also discuss open issues in this research field, for example, the need to incorporate psychological considerations into the design process of user-centric recommender system evaluation studies.

Contribution 2: Illustrating to What Extent Components of the Cognitive Model ACT-R Contribute to Recommendations (2022-2023)

In P5, we discuss transparency aspects of music recommendations generated via ACT-R by illustrating to what extent components of ACT-R have contributed to the generation of recommendation lists. We investigate three ACT-R components described in Section 2.2.2: (i) *the base level learning (BLL)* equation, which describes the “current obsession” of a user (i.e., frequently and recently listened tracks), (ii) the spreading activation (S) component, which describes “current vibes” of a user (i.e., tracks that are similar to the user’s most recently listened track), and (iii) the valuation (V) component, which accounts for “evergreens” of the user (i.e., the user’s most frequently listened tracks, independent of the recency component). Additionally, we analyze a social component (SC) to account for track recommendations “from similar listeners” in the form of user-based CF.

Figure 3.2 shows six recommended tracks for a randomly selected user in our newly created Last.fm dataset sample [195] based on the *LFM-2b* dataset [191, 231]. The heatmap illustrates how the music track recommendations are calculated by showing the relative contribution of these four components to the recommendation score of a track. We see that the components contribute differently to the recommended tracks. For example, for the first track “From the Past Comes the Storms”, the current obsession (*BLL*) of the user is most important, while for the last track “Troops of Doom” solely the social component (SC) contributes to

Recommended Track	Current obsession (BLL)	Current vibes (S)	Evergreens (V)	From similar listeners (SC)
From the Past Comes the Storms	0.471	0.248	0.281	0.000
Escape to the Void	0.306	0.353	0.341	0.000
To the Wall	0.294	0.359	0.347	0.000
R.I.P. (Rest in Pain)	0.264	0.374	0.362	0.000
The Abyss	0.263	0.375	0.362	0.000
Troops of Doom	0.000	0.000	0.000	1.000

Figure 3.2: Heatmap illustrating the relative contribution of three ACT-R components (BLL, S, and V) and one social component (SC) to the recommendation scores of six recommended tracks for a randomly chosen Last.fm user [196].

the recommendation calculation. Based on this, concrete explanations could be derived for all recommendations generated with this model. For example, “this track was recommended because of your *current obsession*”, or “this track was recommended because of *similar listeners*”. We discuss transparency aspects of additional components of ACT-R for music recommendations in a chapter for the “*A Human-centered Perspective of Intelligent Personalized Environments and Systems*” Springer book, which was recently accepted for publication [154].

3.2 Privacy and Limited Preference Information in Recommender Systems

Limited availability of user preference information (e.g., clicks) could be one consequence of data protection initiatives or of the users’ privacy concerns in recommender systems [40, 137, 164, 189, 260] (e.g., users are not willing to share preferences, or to sign in to the system). Thus, we discuss the findings of two studies that address the limited availability of user preference information in the settings of session-based and cold-start recommendations [P6] [P7] (*Contribution 3*). Additionally, we address varying privacy constraints of users in recommender systems (e.g., hiding preferences) [P8], and the accuracy-privacy trade-off of differentially-private recommender systems [P9]. Finally, we survey and categorize the literature on *differential privacy* in collaborative filtering [P10] (*Contribution 4*).

Contribution 3: Addressing Limited User Preference Information in Cold-Start and Session-based Recommendation Settings (2018-2020)

[P6] presents a trust-based CF approach for addressing the user cold-start problem in recommender systems (see Section 2.3.2). Specifically, we aim to exploit implicit and explicit connections between users in trust networks [181] to find the k nearest neighbors and to overcome the limited availability of user preference information in this setting. By employing the idea of *regular equivalence* via *Katz* similarity [109], we do not only find neighbors that share the *same* trust connections, but also neighbors that have *similar* trust connections (i.e., neighbors with similar structural roles in the network). We evaluate our approach using a dataset

from the consumer reviewing portal Epinions [181], which allows users to specify trust connections to other users. We find that our approach outperforms related approaches (e.g., based on Jaccard similarity [58]) in terms of recommendation accuracy for cold-start users. In our follow-up work [71], we employ graph embedding techniques on the trust network of users by evaluating graph embedding methods such as *graph factorization* [21], *DeepWalk* [214], or *Node2Vec* [101] for the user cold-start problem. We find that *Node2Vec* and *DeepWalk* provide the highest recommendation accuracy and user coverage [95] across all methods.

P7 presents our research on using variational autoencoders for session-based job recommendations. Specifically, to provide personalized job recommendations to users in a setting, in which we do not have full user preference histories available, we employ autoencoders to create latent representations of the limited preference information available in the anonymous user sessions (see Section 2.3.2). Our approach recommends jobs within new sessions by employing a k -nearest neighbor approach based on the inferred latent session representations generated via standard autoencoders [36], denoising autoencoders [264], and variational autoencoders [134]. Our evaluation results on session-based job recommendation datasets (e.g., based on XING from the RecSys challenge 2017 [12]) show that our approach based on variational autoencoders provides the most robust results compared to state-of-the-art methods such as *GRU4Rec* [112], *session-KNN* [123], or *sequential session-KNN* [177]. Here, we do not only evaluate recommendation accuracy, but also novelty metrics [262] such as system-based novelty (i.e., how unexplored is the recommended job in general [215]) and session-based novelty (i.e., how surprising is the recommended job for the current user session [279]). To further illustrate the usefulness of variational autoencoders for recommendations, in another paper [230], we utilize them to incorporate a user’s country information into context-aware music recommendations. Specifically, we incorporate the users’ country context into the variational autoencoder architecture via a gating mechanism. Our evaluation results show that our country- and context-aware recommendation approach provides higher recommendation accuracy than related baselines (e.g., variational autoencoders without country information [171]).

Contribution 4: Addressing Users’ Privacy Constraints and the Trade-Off Between Accuracy and Privacy in Recommendations (2021-2023)

P8 studies the robustness of meta matrix factorization (*MetaMF*) against privacy constraints of users in recommender systems. For this, we conduct a reproducibility study of the original *MetaMF* paper [173], and investigate the sensitivity of this approach to the limited availability of user preference information, e.g., when users employ privacy constraints by hiding a certain part of their preferences from the system (see Section 2.3.1). Therefore, we deactivate the *meta learning* [261] component to evaluate the robustness of *MetaMF* against varying privacy constraints. Additionally, we study how users that differ in their profile size (i.e., number of ratings or implicit item preferences) are affected by varying privacy constraints. On the five datasets *Douban* [117], *Hetrec-MovieLens* [50], *MovieLens* 1M [107], *Ciao* [102], and *Jester* [98] (we share the dataset samples via *Zenodo* [202]), we demonstrate that *meta learning* is essential for *MetaMF*’s

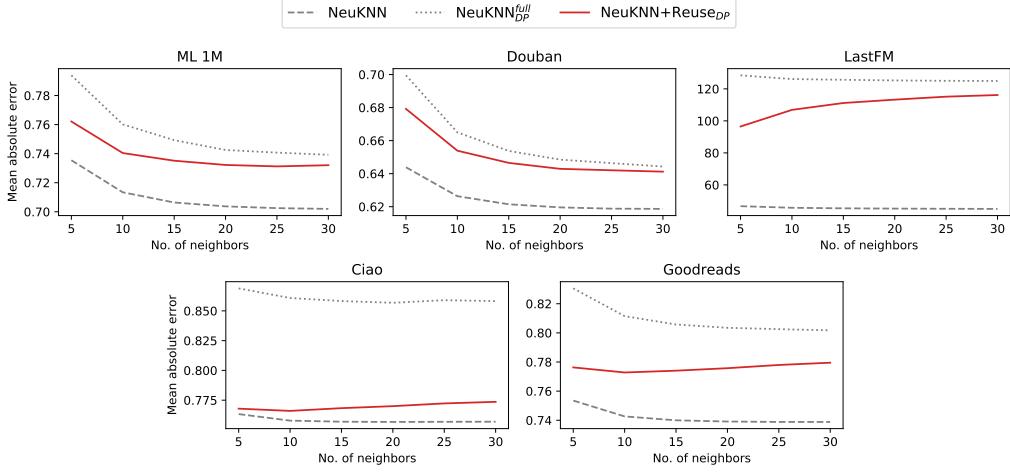


Figure 3.3: Mean absolute error (MAE) of neural-based *KNN* recommender system variants. Our results indicate that combining neighborhood reuse with *differential privacy* (NeuKNN+Reuse_{DP}) yields better accuracy (lower MAE) than neural-based methods that do not apply neighborhood reuse ($\text{NeuKNN}_{DP}^{full}$) [201].

robustness against users' privacy constraints. We also show that users with small profiles can afford a higher degree of privacy than users with large profiles.

[P9] addresses the accuracy-privacy trade-off in differentially-private recommender systems. Specifically, we propose our *ReuseKNN* recommendation approach, which aims to reduce the decrease in accuracy due to the application of *differential privacy* [73, 75] on users' preference information [38]. We achieve this by identifying small but highly reusable neighborhoods for k -nearest neighbor-based recommendation approaches. Therefore, only this small set of users needs to be protected with *differential privacy*, and the majority of the users do not need to be protected, since they are rarely exploited as neighbors, i.e., they have a small privacy risk [175] as defined in Section 2.3.1. We find that with *ReuseKNN*, in the case of a Last.fm dataset sample, only 68.20% of the users need to be protected with *differential privacy*, while a traditional *UserKNN* approach [110] requires the protection of 99.89% of the users. We validate if this also leads to an improved accuracy-privacy trade-off in various recommendation settings. Figure 3.3 shows the recommendation accuracy results of neural-based CF approaches [108] when our neighborhood reuse concept is applied (NeuKNN+Reuse_{DP} , i.e., only vulnerable users are protected), and when it is not applied ($\text{NeuKNN}_{DP}^{full}$, i.e., all users are protected). Additionally, we include a baseline approach without any application of *differential privacy* (NeuKNN , i.e., no users are protected). We see that (i) NeuKNN provides the best accuracy results according to the mean absolute error [269], but without any privacy guarantees, (ii) $\text{NeuKNN}_{DP}^{full}$ provides the worst accuracy results, but with the highest privacy guarantees, and (iii) that our NeuKNN+Reuse_{DP} approach provides a better accuracy-privacy trade-off than the other methods. Additionally, in this work, we outline connections between privacy, and item coverage [111], popularity bias [20], and fairness [79].

Finally, [P10](#) further discusses the accuracy-privacy trade-off in differentially-private recommendations by surveying the literature in this field. Therefore, we identify 26 papers that apply *differential privacy* either (i) to the user representations (e.g., as we do it in [P9](#)), (ii) directly to the recommendation model updates (e.g., when calculating gradients locally), or (iii) after the recommendation model training process (e.g., applying noise to the trained user and item embeddings). We find that these papers address the accuracy-privacy trade-off in three different ways: (i) using auxiliary data to foster recommendation accuracy (e.g., incorporate preferences of other users), (ii) reducing the noise level that is needed (e.g., requiring the minimal amount of noise to still ensure *differential privacy*), and (iii) limit when to apply *differential privacy* (e.g., as we do it in [P9](#)).

3.3 Fairness and Popularity Bias in Recommender Systems

This section discusses our research on fairness and popularity bias in recommender systems. This contains four publications that study popularity bias for user groups that differ in mainstreaminess (i.e., users' inclination towards mainstream content [31]) and gender [P11](#) [P12](#) [P13](#) [P14](#) (*Contribution 5*). This section also describes two papers on understanding popularity bias mitigation and amplification using online and offline evaluation studies [P15](#) [P16](#) (*Contribution 6*). Another journal article analyzes the long-term dynamics of fairness (e.g., *individual* vs. *group* fairness trade-offs) in algorithmic decision support in a labor market setting using agent-based modeling techniques [P17](#) (*Contribution 7*).

Contribution 5: Measuring Popularity Bias for User Groups Differing in Mainstreaminess and Gender (2020-2022)

[P11](#) analyzes the unfairness of popularity bias in music recommendations. Specifically, we reproduce a study by Abdollahpouri et al. [7], in which the authors find that personalized recommendation algorithms in the movie domain are biased towards popular items, and that this popularity bias also leads to the unfair treatment of users with little interest into popular content (see Section 2.4.2). We conduct this reproducibility study in the music domain using a newly created dataset sample [141] gathered from Last.fm. Figure 3.4 shows that our results are in line with the ones of [7] since all evaluated recommendation algorithms tend to favor popular items also in the music domain. In the case of the *Most-Popular* algorithm, as expected, the strongest evidence for popularity bias can be found. In the case of traditional *UserKNN* [110] and *Non-negative Matrix Factorization (NMF)* [178], we also see a positive relationship between item (i.e., music artist) popularity and recommendation frequency. Finally, for *UserKNN* and *NMF*, we find that beyond-mainstream (*BeyMS*) users receive less accurate recommendations than mainstream (*MS*) users (see Figure 3.5a).

In [P12](#), we analyze the unfairly treated *BeyMS* user group in more detail by identifying subgroups of beyond-mainstream music listeners. For this, we create a new dataset termed *LFM-BeyMS*, which contains (among others) audio features

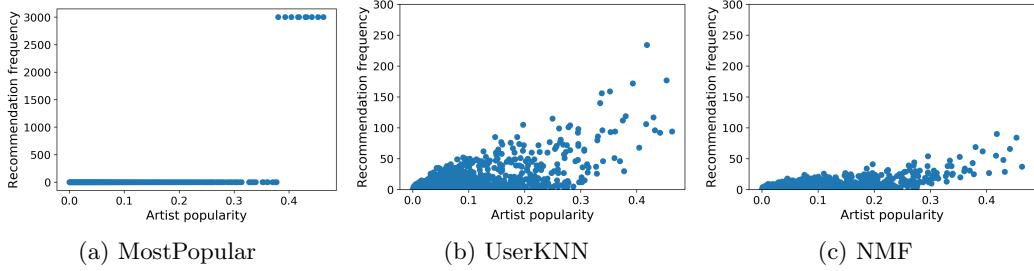


Figure 3.4: Correlation of music artist popularity and recommendation frequency. All three algorithms investigated tend to favor popular music artists [155].

of the music tracks listened to by more than 2,000 *BeyMS* users. Using these audio features and unsupervised clustering techniques, we identify four clusters of beyond-mainstream music and music listeners: (i) U_{folk} , listeners of music with high acousticness such as “folk”, (ii) U_{hard} , listeners of high energy music such as “hardrock”, (iii) U_{ambi} , listeners of music with high acousticness and instrumentalness such as “ambient”, and (iv) U_{elec} , listeners of high energy music with high instrumentalness such as “electronica”. Figure 3.5b shows that there is a substantial difference in recommendation accuracy between these subgroups of *BeyMS* users. While U_{ambi} users, on average, even receive better recommendation accuracy results than *MS* users, U_{hard} users receive the worst recommendation accuracy results. When relating our results to the openness of the subgroups’ users towards music listened to by the other subgroups, we find that U_{ambi} is the most open group, while U_{hard} is the least open group. This is in line with related research [252], which has shown that a user’s openness towards content consumed by other users is positively correlated with recommendation accuracy.

P13 studies if popularity bias in music recommender systems affect users of different genders in the same way. To answer this question, we analyze seven recommendation algorithms, *Random*, *MostPopular*, *ItemKNN* [228], *Sparse Linear Method (SLIM)* [205], *Alternating Least Squares Matrix Factorization (ALS)* [118], *Matrix Factorization with Bayesian Personalized Ranking (BPR)* [222], and *Variational Autoencoder for CF (VAE)* [171], on a Last.fm dataset sample based on the *LFM-2b* dataset [191, 231]. We find that all personalized recommendation algorithms investigated in this study, except for *SLIM*, intensify the popularity bias for female users. Thus, not only user groups differing in mainstreaminess, but also user groups differing in gender are affected differently by popularity bias.

Finally, P14 validates the findings of P11 and P12 in three additional multimedia domains, namely (i) movies (*MovieLens-1M* [107]), (ii) books (*BookCrossing* [280]), and (iii) animes (*MyAnimeList* [219]). For these datasets, we create dataset samples [143] with user groups that differ in their inclination to popular and mainstream content, and analyze popularity bias of various CF-based recommendation algorithms on the levels of items and users. On the item level, we find that the probability of an item to be recommended strongly correlates with the popularity of the item. On the user level, we find that users with the least inclination to popular content also receive the worst recommendation quality.

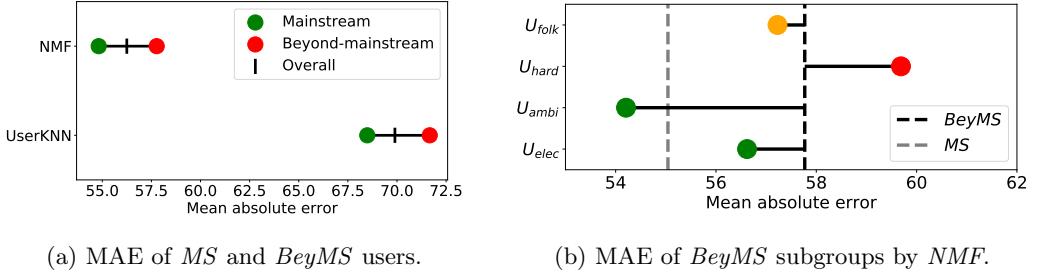


Figure 3.5: (a) Recommendation accuracy measured by the mean absolute error (MAE) of *NMF* and *UserKNN* for mainstream (*MS*) and beyond-mainstream (*BeyMS*) user groups in Last.fm: *BeyMS* users receive a substantially lower recommendation quality (i.e., higher MAE) compared to *MS* users. (b) Comparison of the MAE scores reached by *NMF* for the four *BeyMS* subgroups with the ones reached by *NMF* for *BeyMS* (black dashed line) and *MS* (grey dashed line). There are substantial differences between the subgroups in terms of MAE, especially when comparing U_{hard} with U_{ambi} , i.e., two subgroups differing in their openness to music listened to by users of other subgroups [152].

Contribution 6: Understanding Popularity Bias Mitigation and Amplification in Recommendations (2022-2023)

[P15] presents an online study on popularity bias mitigation (see Section 2.4.2) in a news article recommendation setting. To conduct our online study, we collaborate with *DiePresse*, a popular Austrian online news platform, and discuss the introduction of personalized, content-based news article recommendations into the platform as a replacement for unpersonalized *MostPopular* recommendations. Our content-based recommendation algorithm [63, 176] is based on latent representations of news articles using *Latent Dirichlet Allocation (LDA)* [43]. We conducted our online study in a two-week time window (27th of October 2020 to 9th of November 2020), in which we tracked user preferences (i.e., clicks on news articles) of more than one Million anonymous user sessions, and more than 15,000 signed in (subscribed) users of *DiePresse*. Within our two-week online study, also two significant events happened that could influence the reading behavior of users: (i) the COVID-19 lockdown announcements in Austria on the 31st of October 2020, and (ii) the Vienna terror attack on the 2nd of November 2020.

Figure 3.6 shows the results of our online study in terms of *skewness* and *kurtosis* of the news article popularity distribution (i.e., number of article reads) across the two weeks, and for both user groups (i.e., anonymous and subscribed users). Here, *skewness* measures the asymmetry, and *kurtosis* measures the “tailedness” of the popularity distribution [34]. For both metrics, high values indicate a popularity biased news consumption, which could lead to filter bubble and echo chamber effects [88]. At the beginning of the online study, where *MostPopular* recommendations were shown, we see a large gap between the two user groups: while anonymous users mainly read popular news articles, and thus, are prone to popularity bias, subscribed users show a much more balanced reading behavior. At

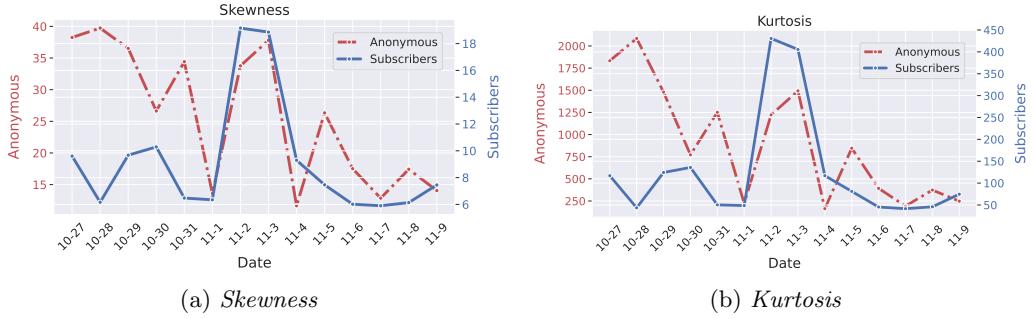


Figure 3.6: Mitigation of popularity bias in news article consumption, measured by (a) *skewness* and (b) *kurtosis* based on the number of article reads for each day of our two-week online study. At the beginning of the study, the *MostPopular* news article recommendations were replaced by personalized, content-based recommendations. We find that popularity bias can be mitigated by introducing personalized news article recommendations in the case of anonymous users [159].

the end of the study, i.e., after two weeks of personalized recommendations, we see a considerably smaller difference between the two user groups, which means that the introduction of personalized, content-based news article recommendations helped to mitigate popularity bias in the case of anonymous users already after two weeks. However, in the case of significant events, e.g., the Vienna terror attack on the 2nd of November 2020, both user groups are mostly interested into popular articles reporting on the particular event. In another work [198], we also find that content-based recommendations can help to mitigate popularity bias in the case of recommendations provided in a data and algorithm sharing platform.

In [P16], we analyze miscalibration [172, 247] and popularity bias amplification (in terms of the popularity lift metric [7, 8]) in music, movie, and anime recommender systems. For this, we extend the *MovieLens 1M* [107], *LFM-1b* [229, 232], and *MyAnimeList* [219] datasets with genre information of the items, and publish these new dataset samples via *Zenodo* [142]. Then we measure accuracy, miscalibration, and popularity bias amplification (i.e., popularity lift) for various recommendation algorithms (e.g., *NMF* [178] and *co-clustering-based CF* [97]), and for user groups differing in their inclination to popular and mainstream content, i.e., (i) *LowPop* (low interest in popular content), (ii) *MedPop* (medium interest in popular content), and (iii) *HighPop* (high interest in popular content). We find that there is a connection between these three metrics, since the *LowPop* user group, which receives the worst recommendation accuracy results, is also the user group, which receives the most miscalibrated and popularity biased recommendations. Finally, we investigate to what extent particular genres contribute to the inconsistency of recommendation performance in terms of miscalibration and popularity bias amplification. We find that there are indeed genres that highly contribute to inconsistent and popularity biased recommendation results. One example is the “*Hentai*” genre in the case of our *MyAnimeList* dataset sample: this is a genre, which is highly popular for a specific user group (i.e., *LowPop*), and unpopular for the other user groups (i.e., *MedPop* and *HighPop*).

Contribution 7: Studying Long-Term Dynamics of Fairness in Algorithmic Decision Support (2022-2023)

[P17] studies the long-term dynamics of fairness in algorithmic decision support (see Section 2.4.1) in a labor market setting [93]. Specifically, we develop and evaluate an agent-based simulation model to investigate the impact of decisions caused by a public employment service that decides which jobseekers receive targeted help using a decision support tool. This tool uses a logistic regression model [270] to classify jobseekers into low- and high-prospects. We use synthetic data that describes a pool of jobseekers with unevenly distributed skills between two groups that differ with respect to a protected attribute. We test two variants of our prediction model: (i) a biased version that augments knowledge about the actual skills of a jobseeker with knowledge about the protected attribute, and (ii) an unbiased version that solely relies on the skills of a jobseeker. Based on the classification into low-prospects and high-prospects, our agent-based simulation model updates the skills of the jobseekers after each iteration accordingly (e.g., a high-prospect receives help, and thus also the skills of this jobseeker increase).

Our results show that there is a trade-off between different long-term fairness goals. On the one hand, when using the biased prediction model, the inequality between the two protected groups is reduced at the end of the simulation. This means, that *statistical parity* in the dataset [30] increases, and that the system is fair from a *group fairness* perspective. However, on the other hand, the number of misclassifications of jobseekers in the unprivileged group increases: some jobseekers are classified as low-prospect mainly because of their sensitive attribute, although they should belong to the high-prospect group. This means that the system is unfair from an *individual fairness* perspective. Although this study was not conducted in the field of recommender systems, we believe that the applied method (i.e., agent-based modeling) could also be of use when studying long-term fairness dynamics of recommender systems. Additionally, our findings with respect to the trade-off between *individual* and *group fairness* are also highly relevant for the research area of fair recommender systems.

3.4 Summary of Contributions and Reproducibility of Research Results

This section summarizes the 7 scientific contributions described in the previous sections. Additionally, the reproducibility of the findings are discussed.

List of Contributions

- 1. Using cognitive models for a transparent design and implementation process of recommender systems (2018-2021):** we propose a tag recommendation approach based on a model of human episodic memory **[P1]**, and two music recommendation approaches based on activation process in human memory **[P2]** **[P3]**. Additionally, we identify three types of psychology-informed recommender systems: (i) cognition-inspired, (ii) personality-aware, and (iii) affect-aware recommender systems **[P4]**.

2. **Illustrating to what extent components of the cognitive model ACT-R contribute to recommendations (2022-2023):** we illustrate to what extent components of ACT-R (e.g., *BLL* or *valuation*) have contributed to the generation of music recommendation lists. Based on this, explanations for the music recommendation could be derived [P5].
3. **Addressing limited user preference information in cold-start and session-based recommendation settings (2018-2020):** we model a user’s trust network using regular equivalence to address the user cold-start problem [P6]. Additionally, we demonstrate the usefulness of variational autoencoders for session-based job recommendations [P7].
4. **Addressing users’ privacy constraints and the accuracy privacy trade-off in recommendations (2021-2023):** we study privacy constraints of users (e.g., hiding preferences) in meta matrix factorization [P8], design a neighborhood reuse approach [P9], and survey the literature for differentially-private collaborative filtering recommender systems [P10].
5. **Measuring popularity bias for user groups differing in mainstreaminess and gender (2020-2022):** we study popularity bias [P11], characteristics of beyond-mainstream users [P12], and differences with respect to users’ gender in music recommendations [P13]. We also show the presence of popularity bias in movie, book, and anime recommendations [P14].
6. **Understanding popularity bias mitigation and amplification in recommendations (2022-2023):** we analyze and mitigate popularity in news article recommender systems [P15], and study to what extent recommendations amplify popularity bias in the music, movie, and anime domains [P16].
7. **Studying long-term dynamics of fairness in algorithmic decision support (2022-2023):** we show the usefulness of agent-based modeling techniques for studying long-term dynamics of algorithmic fairness in a labor market setting. Additionally, we find evidence for the presence of the trade-off between *individual* and *group fairness* in this setting [P17].

Reproducibility of Research Results

To foster the reproducibility of these research results and findings, we provide information on the used source-code and dataset samples in all publications. In cases, in which we create new dataset samples or implement novel recommendation pipelines, we make them freely available via *Zenodo* or *Github*. For example, to implement and evaluate our cognitive-inspired recommendation approaches, we build upon our *TagRec* framework [144, 146], and extend it with music recommendation approaches. Another example is our *Last.fm user group* dataset sample [141] that can be used to study fairness and popularity bias in recommender systems. Additionally, we contribute to reproducibility studies by presenting two papers enlisted in this habilitation in the reproducibility track of the *European Conference on Information Retrieval (ECIR’2020 and ECIR’2021)* [P8] [P11].

A list of the new dataset samples and recommendation pipelines created in the publications that are part of this habilitation is given in the following:

1. The *TagRec* framework [144, 146] used to design, develop, and evaluate cognitive-inspired algorithms for tag and music recommendations: <https://github.com/learning-layers/TagRec>.
2. A *GitHub* repository with the material to generate sequential music recommendations and to illustrate to what extent the components of the cognitive model ACT-R contribute to the generation of music recommendation lists [196]: <https://github.com/hcai-mms/actr>.
3. A dataset sample based on the *LFM-2b* dataset [191, 231] used to generate and evaluate sequential music recommendations [195]. This *Zenodo* repository also contains the pre-calculated embeddings for the *BPR* approach.
4. Source-code and dataset references for using variational autoencoders in the setting of session-based job recommendations [162]: <https://github.com/lacic/session-knn-ae>. This *GitHub* repository also contains implementations of beyond-accuracy evaluation metrics (e.g., diversity and novelty) for session-based recommender systems.
5. A dataset for studying privacy constraints of different users groups using meta matrix factorization [202] accompanied by a *GitHub* repository: <https://github.com/pmuellner/RobustnessOfMetaMF>.
6. The material for the differentially-private *ReuseKNN* [201] recommender system: <https://github.com/pmuellner/ReuseKNN>. This *GitHub* repository also contains the implementation of *Neural CF*, as well as source-code for sampling user preference histories in the datasets.
7. A dataset for studying beyond-mainstream users in music recommender systems [203] accompanied by a *GitHub* repository: <https://github.com/pmuellner/supporttheunderground>. Apart from popularity bias evaluation metrics, this *GitHub* repository contains implementations of unsupervised clustering techniques to analyze audio features of music tracks.
8. Datasets containing different user groups to study fairness and popularity bias in music, movie, book, and anime recommender systems [141, 143]. For calculating calibration-based metrics in these settings, an extended version of these datasets also contains genre information for the items [142].
9. A Python-based pipeline to process the datasets used in [145, 151, 155] for studying fairness and popularity bias in recommender systems: <https://github.com/domkowald/FairRecSys>. This *GitHub* repository can also be used as a basis to develop popularity bias mitigation methods.

By publishing these resources, the author of this habilitation hopes to contribute to reproducible research practices in the field of recommender systems. As mentioned already in Section 2.1, the reproducibility of research results is highly important for being able to track progress in recommender systems research.

Chapter 4

Outlook and Future Research

This chapter gives an outlook into future research directions of this habilitation.

Transparency and Cognitive Models in Recommender Systems

The underlying algorithms of modern recommender systems are often based on purely data-driven machine learning models. Although these approaches provide high accuracy, they are based on principles of artificial intelligence rather than human intelligence. One consequence could be that the logic of these models is not directly understandable by humans, which could lead to non-transparent algorithmic decisions [244]. This habilitation has shown that using psychological theories, and modeling the underlying cognitive processes that describe how humans access information in their memory, is one way to overcome this issue, and at the same time, to generate accurate recommendations (see Section 3.1).

Besides MINERVA2 [113], the cognitive architecture ACT-R [22] provides an excellent basis by formalizing two kinds of human memory: (i) declarative memory, and (ii) procedural memory. The declarative memory corresponds to things that humans know by determining the importance of information chunks, while the procedural memory corresponds to knowledge of how humans do things by defining production rules for making decisions. This habilitation has focused on modeling declarative memory processes for a transparent design process of cognitive-inspired recommender systems. Thus, in future research, I aim to investigate to what extent also the procedural memory module of ACT-R can be used to design recommendation models (e.g., by adapting the *SNIF-ACT* [92] user navigation model). Here, one interesting research question would be if the defined production rules could further contribute to transparency aspects of cognitive-inspired recommender systems. This question could be answered by conducting user studies following well-established procedures in the field (e.g., [138, 210, 255]).

Privacy and Limited Preference Information in Recommender Systems

Privacy is a key requirement for recommender systems, since there are multiple privacy threats to users in these systems. For example, disclosing users' preference information to untrusted third parties [49], or inferring users' sensitive attributes such as gender [277]. Privacy is also related to the issue of limited availability

of user preference information, since users increasingly care about their privacy and may not want to share their preferences with the system [137, 189, 260]. Additionally, initiatives such as the *European General Data Protection Regulation (GDPR)* restrict the use of user preference information to generate recommendations [40, 62]. This habilitation has addressed session-based and cold-start recommendation settings, and the accuracy-privacy trade-off when applying *differential privacy* to the users' preference information (see Section 3.2).

In the future, I plan to not only study the trade-off between accuracy and privacy, but also to investigate other relevant trade-offs between recommendation objectives. This includes the trade-off between privacy and fairness [79]. Here, an interesting research question would be if different user groups are treated differently by the accuracy drops due to privacy-preserving technologies, such as *differential privacy*. For this, related studies from the field of private and fair machine learning (e.g., [29]) could be adapted for recommender systems. Additionally, studying privacy dynamics in recommendations using agent-based simulations would be a promising research direction, as described in our position paper [199] presented in the *SimuRec* workshop of *ACM RecSys* 2021.

Fairness and Popularity Bias in Recommender Systems

Biases in the perception and behavior of humans are captured, reflected, and potentially amplified in recommender systems [55, 91, 163]. The replication of popularity bias is a common issue in collaborative filtering-based recommender systems, which leads to the overrepresentation of popular items in the recommendation lists. The research presented in this habilitation has shown that users with little interest in popular content receive worse recommendation accuracy than users that like to consume popular content. Based on this, these users are treated in an unfair way by the recommender system (see Section 3.3).

In my future research, I plan to work on popularity bias mitigation methods to reduce the accuracy differences between the user groups, and with this, increase the fairness in the system. For this, not only technical debiasing methods (e.g., in- or post-processing [9]), but also novel multidisciplinary approaches using models from psychology and physics should be developed. For the former, ACT-R [22] could be a promising basis to build strongly personalized user models, and for the latter, techniques from physics-informed machine learning could be transferred to fairness problems, as described in our recent *arXiv* pre-print [236].

Reproducibility Aspects of this Habilitation

I want to highlight the importance of reproducibility for the research field of recommender systems [32, 86]. This habilitation has provided several resources to foster the reproducibility of the presented research results (see Section 3.4). In the future, I want to further contribute to the reproducibility of machine learning research in general, and recommender systems research in particular, by discussing barriers and best practices as outlined in our recent *arXiv* pre-print [241].

Finally, I hope that the scientific results and findings of this habilitation contribute to advancing research on the trustworthiness of recommender systems.

Bibliography

- [1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. Multistakeholder recommendation: Survey and research directions. *User Modeling and User-Adapted Interaction*, 30:127–158, 2020.
- [2] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. Controlling popularity bias in learning-to-rank recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 42–46, 2017.
- [3] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. Recommender systems as multistakeholder environments. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 347–348, 2017.
- [4] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. Managing popularity bias in recommender systems with personalized re-ranking. In *The Thirty-Second International FLAIRS Conference*, 2019.
- [5] Himan Abdollahpouri, Edward C Malthouse, Joseph A Konstan, Bamshad Mobasher, and Jeremy Gilbert. Toward the next generation of news recommender systems. In *Companion Proceedings of the Web Conference 2021*, pages 402–406, 2021.
- [6] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. The impact of popularity bias on fairness and calibration in recommendation. *arXiv preprint arXiv:1910.05755*, 2019.
- [7] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. The unfairness of popularity bias in recommendation. In *RMSE Workshop co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019)*, 2019.
- [8] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. The connection between popularity bias, calibration, and fairness in recommendation. In *Fourteenth ACM Conference on Recommender Systems*, pages 726–731, 2020.
- [9] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, Bamshad Mobasher, and Edward Malthouse. User-centered evaluation of popularity

- bias in recommender systems. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, pages 119–129, 2021.
- [10] Fabian Abel. We know where you should work next summer: Job recommendations. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 230–230, 2015.
 - [11] Fabian Abel, András Benczúr, Daniel Kohlsdorf, Martha Larson, and Róbert Pálovics. Recsys challenge 2016: Job recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 425–426, 2016.
 - [12] Fabian Abel, Yashar Deldjoo, Mehdi Elahi, and Daniel Kohlsdorf. Recsys challenge 2017: Offline and online evaluation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 372–373, 2017.
 - [13] Panagiotis Adamopoulos and Alexander Tuzhilin. On unexpectedness in recommender systems: Or how to better expect the unexpected. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(4):1–32, 2014.
 - [14] Gediminas Adomavicius, Konstantin Bauman, Alexander Tuzhilin, and Moshe Unger. Context-aware recommender systems: From foundations to recent developments context-aware recommender systems. In *Recommender Systems Handbook*, pages 211–250. Springer, 2021.
 - [15] Gediminas Adomavicius, Jesse C Bockstedt, Shawn P Curley, and Jingjing Zhang. Do recommender systems manipulate consumer preferences? a study of anchoring effects. *Information Systems Research*, 24(4):956–975, 2013.
 - [16] Gediminas Adomavicius and YoungOk Kwon. Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):896–911, 2011.
 - [17] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, and Alexander Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.
 - [18] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
 - [19] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In *Recommender Systems Handbook*, pages 217–253. Springer, 2010.
 - [20] Abdul Basit Ahanger, Syed Wajid Aalam, Muzafar Rasool Bhat, and Assif Assad. Popularity bias in recommender systems - a review. In *International Conference on Emerging Technologies in Computer Engineering*, pages 431–444. Springer, 2022.

- [21] Amr Ahmed, Nino Shervashidze, Shravan Narayananurthy, Vanja Josifovski, and Alexander J Smola. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 37–48, 2013.
- [22] John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. An integrated theory of the mind. *Psychological Review*, 111(4):25 pages, 2004.
- [23] John R Anderson, Michael Matessa, and Christian Lebiere. Act-r: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12(4):439–462, 1997.
- [24] John R Anderson and Lael J Schooler. Reflections of the environment in memory. *Psychological Science*, 2(6):396–408, 1991.
- [25] Vito Walter Anelli, Yashar Deldjoo, Tommaso Di Noia, Antonio Ferrara, and Fedelucio Narducci. Federank: User controlled feedback with federated recommender systems. In *Advances in Information Retrieval: 43rd European Conference on IR Research (ECIR 2021)*, pages 32–47. Springer, 2021.
- [26] Arda Antikacioglu and R Ravi. Post processing recommender systems for diversity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 707–716, 2017.
- [27] Müslüm Atas, Alexander Felfernig, Seda Polat-Erdeniz, Andrei Popescu, Thi Ngoc Trang Tran, and Mathias Uta. Towards psychology-aware preference construction in recommender systems: Overview and research issues. *Journal of Intelligent Information Systems*, 57:467–489, 2021.
- [28] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM Press New York, 1999.
- [29] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in Neural Information Processing Systems*, 32, 2019.
- [30] Solon Barocas and Andrew D Selbst. Big data’s disparate impact. *California Law Review*, pages 671–732, 2016.
- [31] Christine Bauer and Markus Schedl. Global and country-specific mainstreaminess measures: Definitions, analysis, and usage for improving personalized music recommendation systems. *PloS One*, 14(6):e0217389, 2019.
- [32] Joeran Beel, Corinna Breitinger, Stefan Langer, Andreas Lommatzsch, and Bela Gipp. Towards reproducibility in recommender-systems research. *User Modeling and User-Adapted Interaction*, 26:69–101, 2016.
- [33] Ghazaleh Beigi and Huan Liu. A survey on privacy in social media: Identification, mitigation, and applications. *ACM Transactions on Data Science*, 1(1):1–38, 2020.

- [34] Alejandro Bellogín, Pablo Castells, and Iván Cantador. Statistical biases in information retrieval metrics for recommender systems. *Information Retrieval Journal*, 20:606–634, 2017.
- [35] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise Reduction in Speech Processing*, pages 37–40. Springer, 2009.
- [36] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems*, 19, 2006.
- [37] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 1:42, 2018.
- [38] Shlomo Berkovsky, Tsvi Kuflik, and Francesco Ricci. The impact of data obfuscation on the accuracy of collaborative filtering. *Expert Systems with Applications*, 39(5):5033–5042, 2012.
- [39] James R Bettman and C Whan Park. Effects of prior knowledge and experience and phase of the choice process on consumer decision processes: A protocol analysis. *Journal of Consumer Research*, 7(3):234–248, 1980.
- [40] Asia J Biega, Peter Potash, Hal Daumé, Fernando Diaz, and Michèle Finck. Operationalizing the legal principle of data minimization for personalization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 399–408, 2020.
- [41] Vincent Bindschaedler, Reza Shokri, and Carl A Gunter. Plausible deniability for privacy-preserving data synthesis. *Proceedings of the VLDB Endowment*, 10(5), 2017.
- [42] Reuben Binns. On the apparent conflict between individual and group fairness. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 514–524, 2020.
- [43] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.
- [44] Erik Brynjolfsson, Yu Jeffrey Hu, and Michael D Smith. From niches to riches: Anatomy of the long tail. *Sloan Management Review*, 47(4):67–71, 2006.
- [45] Robin Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-adapted Interaction*, 12:331–370, 2002.
- [46] Robin Burke. Hybrid web recommender systems. *The Adaptive Web: Methods and Strategies of Web Personalization*, pages 377–408, 2007.
- [47] Robin Burke, Alexander Felfernig, and Mehmet H Göker. Recommender systems: An overview. *AI Magazine*, 32(3):13–18, 2011.

- [48] Robin Burke, Nasim Sonboli, and Aldo Ordonez-Gauger. Balanced neighborhoods for multi-sided fairness in recommendation. In *Conference on Fairness, Accountability and Transparency*, pages 202–214. PMLR, 2018.
- [49] Joseph A Calandrino, Ann Kilzer, Arvind Narayanan, Edward W Felten, and Vitaly Shmatikov. You might also like: Privacy risks of collaborative filtering. In *Proceedings of 2011 IEEE Symposium on Security and Privacy*, pages 231–246. IEEE, 2011.
- [50] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *Proceedings of the Fifth ACM Conference on Recommender Systems*, pages 387–388, 2011.
- [51] Pablo Castells, Neil Hurley, and Saul Vargas. Novelty and diversity in recommender systems. In *Recommender Systems Handbook*, pages 603–646. Springer, 2021.
- [52] Pablo Castells and Dietmar Jannach. Recommender systems: A primer. *arXiv preprint arXiv:2302.02579*, 2023.
- [53] Pablo Castells and Alistair Moffat. Offline recommender system evaluation: Challenges and new directions. *AI Magazine*, 43(2):225–238, 2022.
- [54] Óscar Celma and Perfecto Herrera. A new approach to evaluating novel recommendations. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, pages 179–186, 2008.
- [55] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems*, 41(3):1–39, 2023.
- [56] Li Chen, Yonghua Yang, Ningxia Wang, Keping Yang, and Quan Yuan. How serendipity improves user satisfaction with recommendations? a large-scale user evaluation. In *The World Wide Web Conference*, pages 240–250, 2019.
- [57] Xiaocong Chen, Lina Yao, Julian McAuley, Guanglin Zhou, and Xianzhi Wang. Deep reinforcement learning in recommender systems: A survey and new perspectives. *Knowledge-Based Systems*, 264:110335, 2023.
- [58] Pern Hui Chia and Georgios Pitsilis. Exploring the use of explicit trust links for filtering recommenders: A study on opinions.com. *Journal of Information Processing*, 19:332–344, 2011.
- [59] European Commission. Proposal for a regulation laying down harmonised rules on artificial intelligence (eu ai act), url: <https://eur-lex.europa.eu/legal-content/en/txt/?uri=celex:52021pc0206>, 2021. Accessed on November 22nd, 2023.

- [60] Paolo Cremonesi and Dietmar Jannach. Progress in recommender systems research: Crisis? what crisis? *AI Magazine*, 42(3):43–54, 2021.
- [61] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, pages 39–46, 2010.
- [62] Rachel Cummings and Deven Desai. The role of differential privacy in gdpr compliance. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT’18)*, page 20, 2018.
- [63] Marco De Gemmis, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. Semantics-aware content-based recommender systems. *Recommender Systems Handbook*, pages 119–159, 2015.
- [64] Gabriel de Souza Pereira Moreira, Felipe Ferreira, and Adilson Marques da Cunha. News session-based recommendations using deep neural networks. In *Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems*, pages 15–23, 2018.
- [65] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogín, Alessandro Difonzo, and Dario Zanzonelli. Fairness in recommender systems: Research landscape and future directions. *User Modeling and User-Adapted Interaction*, pages 1–50, 2023.
- [66] Yashar Deldjoo, Markus Schedl, Paolo Cremonesi, and Gabriella Pasi. Recommender systems leveraging multimedia content. *ACM Computing Surveys (CSUR)*, 53(5):1–38, 2020.
- [67] Sebastian Dennerlein, Dieter Theiler, Peter Marton, Patricia Santos Rodriguez, John Cook, Stefanie Lindstaedt, and Elisabeth Lex. Knowbrain: An online social knowledge repository for informal workplace learning. In *10th European Conference on Technology Enhanced Learning (EC-TEL 2015)*, pages 509–512. Springer, 2015.
- [68] Christian Desrosiers and George Karypis. A comprehensive survey of neighborhood-based recommendation methods. *Recommender Systems Handbook*, pages 107–144, 2010.
- [69] Tommaso Di Noia, Nava Tintarev, Panagiota Fatourou, and Markus Schedl. Recommender systems under european ai regulations. *Communications of the ACM*, 65(4):69–73, 2022.
- [70] Gideon Dror, Noam Koenigstein, Yehuda Koren, and Markus Weimer. The yahoo! music dataset and kdd-cup’11. In *Proceedings of KDD Cup 2011*, pages 3–18. PMLR, 2012.
- [71] Tomislav Duricic, Hussain Hussain, Emanuel Lacic, Dominik Kowald, Denis Helic, and Elisabeth Lex. Empirical comparison of graph embeddings for trust-based collaborative filtering. In *Foundations of Intelligent Systems: 25th International Symposium, ISMIS 2020*, pages 181–191. Springer, 2020.

- [72] Tomislav Duricic, Emanuel Lacic, Dominik Kowald, and Elisabeth Lex. Trust-based collaborative filtering: Tackling the cold start problem using regular equivalence. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 446–450, 2018.
- [73] Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.
- [74] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226, 2012.
- [75] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.
- [76] Magdalini Eirinaki, Jerry Gao, Iraklis Varlamis, and Konstantinos Tserpes. Recommender systems for large-scale social networks: A review of challenges and solutions. *Future Generation Computer Systems*, 78:413–418, 2018.
- [77] Magdalini Eirinaki, Malamati D Louta, and Iraklis Varlamis. A trust-aware system for personalized user recommendations in social networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(4):409–421, 2013.
- [78] Michael D Ekstrand, Anubrata Das, Robin Burke, Fernando Diaz, et al. Fairness in information access systems. *Foundations and Trends in Information Retrieval*, 16(1-2):1–177, 2022.
- [79] Michael D Ekstrand, Rezvan Joshaghani, and Hoda Mehrpouyan. Privacy for all: Ensuring fair and equitable privacy protections. In *Conference on Fairness, Accountability and Transparency*, pages 35–47. PMLR, 2018.
- [80] Michael D Ekstrand, John T Riedl, Joseph A Konstan, et al. Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.
- [81] Michael D Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In *Conference on Fairness, Accountability and Transparency*, pages 172–186. PMLR, 2018.
- [82] Mehdi Elahi, Danial Khosh Kholgh, Mohammad Sina Kiarostami, Sorush Saghari, Shiva Parsa Rad, and Marko Tkalcic. Investigating the impact of recommender systems on user-based and item-based popularity bias. *Information Processing & Management*, 58(5):102655, 2021.

- [83] Wenqi Fan, Xiangyu Zhao, Xiao Chen, Jingran Su, Jingtong Gao, Lin Wang, Qidong Liu, Yiqi Wang, Han Xu, Lei Chen, et al. A comprehensive survey on trustworthy recommender systems. *arXiv preprint arXiv:2209.10117*, 2022.
- [84] Wenqi Fan, Xiangyu Zhao, Lin Wang, Xiao Chen, Jingtong Gao, Qidong Liu, and Shijie Wang. Trustworthy recommender systems: Foundations and frontiers. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5796–5797, 2023.
- [85] Alexander Felfernig, Gerhard Friedrich, Bartosz Gula, Martin Hitz, Thomas Kruggel, Gerhard Leitner, Rudolf Melcher, Daniela Riepan, Sabine Strauss, Erich Teppan, et al. Persuasive recommendation: Serial position effects in knowledge-based recommender systems. In *Persuasive Technology: Second International Conference on Persuasive Technology (PERSUASIVE 2007)*, pages 283–294. Springer, 2007.
- [86] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Transactions on Information Systems (TOIS)*, 39(2):1–49, 2021.
- [87] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM conference on recommender systems*, pages 101–109, 2019.
- [88] Seth Flaxman, Sharad Goel, and Justin M Rao. Filter bubbles, echo chambers, and online news consumption. *Public Opinion Quarterly*, 80(S1):298–320, 2016.
- [89] Arik Friedman, Shlomo Berkovsky, and Mohamed Ali Kaafar. A differential privacy framework for matrix factorization recommender systems. *User Modeling and User-Adapted Interaction*, 26:425–458, 2016.
- [90] Arik Friedman, Bart P Knijnenburg, Kris Vanhecke, Luc Martens, and Shlomo Berkovsky. Privacy aspects of recommender systems. *Recommender Systems Handbook*, pages 649–688, 2015.
- [91] Batya Friedman and Helen Nissenbaum. Bias in computer systems. *ACM Transactions on Information Systems (TOIS)*, 14(3):330–347, 1996.
- [92] Wai-Tat Fu and Peter Pirolli. Snif-act: A cognitive model of user navigation on the world wide web. *Human–Computer Interaction*, 22(4):355–412, 2007.
- [93] Simon Gächter and Ernst Fehr. Fairness in the labour market: A survey of experimental results. In *Surveys in Experimental Economics: Bargaining, Cooperation and Election Stock Markets*, pages 95–132. Springer, 2002.

- [94] Florent Garcin, Boi Faltings, Olivier Donatsch, Ayar Alazzawi, Christophe Bruttin, and Amr Huber. Offline and online evaluation of news recommender systems at swissinfo.ch. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 169–176, 2014.
- [95] Mouzhi Ge, Carla Delgado-Battenfeld, and Dietmar Jannach. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, pages 257–260, 2010.
- [96] Yingqiang Ge, Shuchang Liu, Zuohui Fu, Juntao Tan, Zelong Li, Shuyuan Xu, Yunqi Li, Yikun Xian, and Yongfeng Zhang. A survey on trustworthy recommender systems. *arXiv preprint arXiv:2207.12515*, 2022.
- [97] Thomas George and Srujana Merugu. A scalable collaborative filtering framework based on co-clustering. In *Fifth IEEE International Conference on Data Mining (ICDM’05)*, pages 4–pp. IEEE, 2005.
- [98] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4:133–151, 2001.
- [99] Gustavo González, Beatriz López, and Josep Lluís de la Rosa. The emotional factor: An innovative approach to user modelling for recommender systems. In *Workshop on Recommendation and Personalization in e-Commerce*, pages 90–99, 2002.
- [100] Irwin Greenberg. An analysis of the eeocc “four-fifths” rule. *Management Science*, 25(8):762–769, 1979.
- [101] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864, 2016.
- [102] Guibing Guo, Jie Zhang, Daniel Thalmann, and Neil Yorke-Smith. Etaf: An extended trust antecedents framework for trust prediction. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 540–547. IEEE, 2014.
- [103] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 1, 2015.
- [104] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. A novel recommendation model regularized with user trust and item ratings. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1607–1620, 2016.
- [105] Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *IEEE Transactions on Knowledge and Data Engineering*, 25(7):1445–1459, 2012.

- [106] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29, 2016.
- [107] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(4):1–19, 2015.
- [108] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182, 2017.
- [109] Denis Helic. Regular equivalence in informed network search. In *37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1088–1093. IEEE, 2014.
- [110] Jonathan L Herlocker, Joseph A Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 230–237, 1999.
- [111] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.
- [112] Balázs Hidasi and Alexandros Karatzoglou. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 843–852, 2018.
- [113] Douglas L Hintzman. Minerva 2: A simulation model of human memory. *Behavior Research Methods, Instruments, & Computers*, 16(2):96–101, 1984.
- [114] Douglas L Hintzman. Schema abstraction in a multiple-trace memory model. *Psychological Review*, 93(4):411, 1986.
- [115] Douglas L Hintzman. Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review*, 95(4):528, 1988.
- [116] Frank Hopfgartner, Torben Brodt, Jonas Seiler, Benjamin Kille, Andreas Lommatzsch, Martha Larson, Roberto Turrin, and András Serény. Benchmarking news recommendations: The clef newsreel use case. In *ACM SIGIR Forum*, volume 49, 2, pages 129–136. ACM New York, NY, USA, 2016.
- [117] Longke Hu, Aixin Sun, and Yong Liu. Your neighbors affect your ratings: On geographical neighborhood influence to rating prediction. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 345–354, 2014.

- [118] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pages 263–272. IEEE, 2008.
- [119] Dietmar Jannach and Gediminas Adomavicius. Recommendations with a purpose. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 7–10, 2016.
- [120] Dietmar Jannach and Christine Bauer. Escaping the mcnamara fallacy: Towards more impactful recommender systems research. *AI Magazine*, 41(4):79–95, 2020.
- [121] Dietmar Jannach and Michael Jugovac. Measuring the business value of recommender systems. *ACM Transactions on Management Information Systems (TMIS)*, 10(4):1–23, 2019.
- [122] Dietmar Jannach, Lukas Lerche, and Markus Zanker. Recommending based on implicit feedback. In *Social Information Access: Systems and Technologies*, pages 510–569. Springer, 2018.
- [123] Dietmar Jannach and Malte Ludewig. When recurrent neural networks meet the neighborhood for session-based recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 306–310, 2017.
- [124] Dietmar Jannach, Massimo Quadrana, and Paolo Cremonesi. Session-based recommender systems. In *Recommender Systems Handbook*, pages 301–334. Springer, 2022.
- [125] Dietmar Jannach, Paul Resnick, Alexander Tuzhilin, and Markus Zanker. Recommender systems — beyond matrix completion. *Communications of the ACM*, 59(11):94–102, 2016.
- [126] Dietmar Jannach and Markus Zanker. Impact and value of recommender systems. *Recommender Systems Handbook*, 2022.
- [127] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender systems: An introduction*. Cambridge University Press, 2010.
- [128] Robert Jäschke, Leandro Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. Tag recommendations in folksonomies. In *11th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2007)*, pages 506–514. Springer, 2007.
- [129] Arjan JP Jeckmans, Michael Beye, Zekeriya Erkin, Pieter Hartel, Reginald L Lagendijk, and Qiang Tang. Privacy in recommender systems. In *Social Media Retrieval*, pages 263–281. Springer, 2013.
- [130] Yucheng Jin, Nava Tintarev, and Katrien Verbert. Effects of personal characteristics on music recommender systems with different levels of controllability. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 13–21, 2018.

- [131] Ion Juvina, Othalia Larue, and Alexander Hough. Modeling valuation and core affect in a cognitive architecture: The impact of valence and arousal on memory and decision-making. *Cognitive Systems Research*, 48:4–24, 2018.
- [132] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 197–206. IEEE, 2018.
- [133] Krishnaram Kenthapadi, Benjamin Le, and Ganesh Venkataraman. Personalized job recommendation system at linkedin: Practical challenges and lessons learned. In *Proceedings of the Eleventh ACM conference on Recommender Systems*, pages 346–347, 2017.
- [134] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *Stat*, 1050:1, 2014.
- [135] Anastasiia Klimashevskaya, Mehdi Elahi, Dietmar Jannach, Christoph Trattner, and Lars Skjærven. Mitigating popularity bias in recommendation: Potential and limits of calibration approaches. In *International Workshop on Algorithmic Bias in Search and Recommendation*, pages 82–90. Springer, 2022.
- [136] Anastasiia Klimashevskaya, Dietmar Jannach, Mehdi Elahi, and Christoph Trattner. A survey on popularity bias in recommender systems. *arXiv preprint arXiv:2308.01118*, 2023.
- [137] Bart P Knijnenburg and Alfred Kobsa. Making decisions about privacy: Information disclosure in context-aware recommender systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 3(3):1–23, 2013.
- [138] Bart P Knijnenburg, Martijn C Willemse, Zeno Gantner, Hakan Soncu, and Chris Newell. Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction*, 22:441–504, 2012.
- [139] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [140] Pigi Kouki, Shobeir Fakhraei, James Foulds, Magdalini Eirinaki, and Lise Getoor. Hyper: A flexible and extensible probabilistic framework for hybrid recommender systems. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 99–106, 2015.
- [141] Dominik Kowald. Lfm user groups, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.3475975>, 2019.
- [142] Dominik Kowald. Datasets to evaluate accuracy, miscalibration and popularity lift in recommendations, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.7428435>, 2022.
- [143] Dominik Kowald. Fair recsys datasets, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.6123879>, 2022.

- [144] Dominik Kowald, Simone Kopeinik, and Elisabeth Lex. The tagrec framework as a toolkit for the development of tag-based recommender systems. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 23–28, 2017.
- [145] Dominik Kowald and Emanuel Lacic. Popularity bias in collaborative filtering-based multimedia recommender systems. In *Advances in Bias and Fairness in Information Retrieval, BIAS 2022*, pages 1–11. Springer, 2022.
- [146] Dominik Kowald, Emanuel Lacic, and Christoph Trattner. Tagrec: Towards a standardized tag recommender benchmarking framework. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media*, pages 305–307, 2014.
- [147] Dominik Kowald and Elisabeth Lex. Evaluating tag recommender algorithms in real-world folksonomies: A comparative study. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 265–268, 2015.
- [148] Dominik Kowald and Elisabeth Lex. The influence of frequency, recency and semantic context on the reuse of tags in social tagging systems. In *Proceedings of the 27th ACM Conference on Hypertext and Social Media*, pages 237–242, 2016.
- [149] Dominik Kowald*, Elisabeth Lex*, and Markus Schedl. Modeling artist preferences for personalized music recommendations. In *Late-Breaking-Results of the 20th annual conference of the International Society for Music Information Retrieval, ISMIR ’19*, 2019, *equal contribution.
- [150] Dominik Kowald*, Elisabeth Lex*, and Markus Schedl. Utilizing human memory processes to model genre preferences for personalized music recommendations. In *4th Workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory*. Association of Computing Machinery, 2020, *equal contribution.
- [151] Dominik Kowald*, Gregor Mayr*, Markus Schedl, and Elisabeth Lex. A study on accuracy, miscalibration, and popularity bias in recommendations. In *Advances in Bias and Fairness in Information Retrieval, BIAS 2023*, pages 1–16. Springer, 2023, *equal contribution.
- [152] Dominik Kowald, Peter Muellner, Eva Zangerle, Christine Bauer, Markus Schedl, and Elisabeth Lex. Support the underground: Characteristics of beyond-mainstream music listeners. *EPJ Data Science*, 10(1):1–26, 2021.
- [153] Dominik Kowald, Subhash Chandra Pujari, and Elisabeth Lex. Temporal effects on hashtag reuse in twitter: A cognitive-inspired hashtag recommendation approach. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1401–1410, 2017.

- [154] Dominik Kowald, Markus Reiter-Haas, Simone Kopeinik, Markus Schedl, and Elisabeth Lex. Transparent music preference modeling and recommendation with a model of human memory theory. In *A Human-centered Perspective of Intelligent Personalized Environments and Systems*. Springer, 2023.
- [155] Dominik Kowald, Markus Schedl, and Elisabeth Lex. The unfairness of popularity bias in music recommendation: A reproducibility study. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020*, pages 35–42. Springer, 2020.
- [156] Dominik Kowald, Paul Seitlinger, Tobias Ley, and Elisabeth Lex. The impact of semantic context cues on the user acceptance of tag recommendations: An online study. In *Companion Proceedings of the Web Conference 2018*, pages 1–2, 2018.
- [157] Mark A Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChe Journal*, 37(2):233–243, 1991.
- [158] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. *Advances in Neural Information Processing Systems*, 30, 2017.
- [159] Emanuel Lacic, Leon Fadljevic, Franz Weissenboeck, Stefanie Lindstaedt, and Dominik Kowald. What drives readership? an online study on user interface types and popularity bias mitigation in news article recommendations. In *European Conference on Information Retrieval*, pages 172–179. Springer, 2022.
- [160] Emanuel Lacic, Dominik Kowald, Markus Reiter-Haas, Valentin Slawicek, and Elisabeth Lex. Beyond accuracy optimization: On the value of item embeddings for student job recommendations. In *International Workshop on Multi-dimensional Information Fusion for User Modeling and Personalization (IFUP’2018) co-located with the 11th ACM International Conference on Web Search and Data Mining (WSDM’2018)*, 2017.
- [161] Emanuel Lacic, Dominik Kowald, Matthias Traub, Granit Luzhnica, Jörg Peter Simon, and Elisabeth Lex. Tackling cold-start users in recommender systems with indoor positioning systems. In *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 2015.
- [162] Emanuel Lacic, Markus Reiter-Haas, Dominik Kowald, Manoj Reddy Dareddy, Junghoo Cho, and Elisabeth Lex. Using autoencoders for session-based job recommendations. *User Modeling and User-Adapted Interaction*, 30:617–658, 2020.
- [163] Anja Lambrecht and Catherine Tucker. Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads. *Management Science*, 65(7):2966–2981, 2019.

- [164] Martha Larson, Allesandro Zito, Babak Loni, Paolo Cremonesi, et al. Towards minimal necessary data: The case for analyzing training data requirements of recommender algorithms. In *Proceedings of the 2017 FATREC Workshop on Responsible Recommendation*, pages 1–6, 2017.
- [165] Angela Y Lee. Effects of implicit memory on memory-based versus stimulus-based brand choice. *Journal of Marketing Research*, 39(4):440–454, 2002.
- [166] Oleg Lesota, Alessandro Melchiorre, Navid Rekabsaz, Stefan Brandl, Dominik Kowald, Elisabeth Lex, and Markus Schedl. Analyzing item popularity bias of music recommender systems: are different genders equally affected? In *Proceedings of the 15th ACM Conference on Recommender Systems*, pages 601–606, 2021.
- [167] Elisabeth Lex and Dominik Kowald. The impact of time on hashtag reuse in twitter: A cognitive-inspired hashtag recommendation approach. In *Proceedings of the 49th GI Annual Conference*, INFORMATIK ’19, 2019.
- [168] Elisabeth Lex*, Dominik Kowald*, and Markus Schedl. Modeling popularity and temporal drift of music genre preferences. *Transactions of the International Society for Music Information Retrieval*, 3(1), 2020, *equal contribution.
- [169] Elisabeth Lex, Dominik Kowald, Paul Seitlinger, Thi Ngoc Trang Tran, Alexander Felfernig, Markus Schedl, et al. Psychology-informed recommender systems. *Foundations and Trends in Information Retrieval*, 15(2):134–242, 2021.
- [170] Zihao Li, Xianzhi Wang, Chao Yang, Lina Yao, Julian McAuley, and Guandong Xu. Exploiting explicit and implicit item relationships for session-based recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 553–561, 2023.
- [171] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 World Wide Web Conference*, pages 689–698, 2018.
- [172] Kun Lin, Nasim Sonboli, Bamshad Mobasher, and Robin Burke. Calibration in collaborative filtering recommender systems: A user-centered analysis. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media*, HT ’20, page 197–206, New York, NY, USA, 2020. Association for Computing Machinery.
- [173] Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Dongxiao Yu, Jun Ma, Maarten de Rijke, and Xiuzhen Cheng. Meta matrix factorization for federated rating predictions. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 981–990, 2020.

- [174] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.
- [175] Kun Liu and Evimaria Terzi. A framework for computing the privacy scores of users in online social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(1):1–30, 2010.
- [176] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. *Recommender Systems Handbook*, pages 73–105, 2010.
- [177] Malte Ludewig and Dietmar Jannach. Evaluation of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction*, 28:331–390, 2018.
- [178] Xin Luo, Mengchu Zhou, Yunni Xia, and Qingsheng Zhu. An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Transactions on Industrial Informatics*, 10(2):1273–1284, 2014.
- [179] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. Feedback loop and bias amplification in recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2145–2148, 2020.
- [180] Leandro Balby Marinho and Lars Schmidt-Thieme. Collaborative tag recommendations. In *Data Analysis, Machine Learning and Applications: Proceedings of the 31st Annual Conference of the Gesellschaft für Klassifikation eV*, pages 533–540. Springer, 2008.
- [181] Paolo Massa and Paolo Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM Conference on Recommender Systems*, pages 17–24, 2007.
- [182] Judith Masthoff. Group modeling: Selecting a sequence of television items to suit a group of viewers. *Personalized Digital Television: Targeting Programs to Individual Viewers*, pages 93–141, 2004.
- [183] Judith Masthoff. The pursuit of satisfaction: Affective state in group recommender systems. In *International Conference on User Modeling*, pages 297–306. Springer, 2005.
- [184] Judith Masthoff. Group recommender systems: Combining individual models. In *Recommender Systems Handbook*, pages 677–702. Springer, 2010.
- [185] Judith Masthoff and Amra Delic. Group recommender systems: Beyond preference aggregation. *Recommender Systems Handbook*, page 381, 2022.
- [186] Judith Masthoff and Albert Gatt. In pursuit of satisfaction and the prevention of embarrassment: Affective state in group recommender systems. *User Modeling and User-Adapted Interaction*, 16:281–319, 2006.

- [187] Sean M McNee, John Riedl, and Joseph A Konstan. Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems*, pages 1097–1101, 2006.
- [188] Frank McSherry and Ilya Mironov. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 627–636, 2009.
- [189] AKM Nuhil Mehdy, Michael D Ekstrand, Bart P Knijnenburg, and Hoda Mehrpouyan. Privacy as a planned behavior: Effects of situational factors on privacy perceptions and plans. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, pages 169–178, 2021.
- [190] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.
- [191] Alessandro B Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. Investigating gender fairness of recommendation algorithms in the music domain. *Information Processing & Management*, 58(5):102666, 2021.
- [192] Alessandro B Melchiorre, Eva Zangerle, and Markus Schedl. Personality bias of music recommendation algorithms. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pages 533–538, 2020.
- [193] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.
- [194] Nuno Moniz, Luís Torgo, Magdalini Eirinaki, and Paula Branco. A framework for recommendation of highly popular news lacking social feedback. *New Generation Computing*, 35:417–450, 2017.
- [195] Marta Moscati, Christian Wallmann, Markus Reiter-Haas, Dominik Kowald, Elisabeth Lex, and Markus Schedl. Files for integrating the act-r framework with collaborative filtering for explainable sequential music recommendation, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.7923580>, 2023.
- [196] Marta Moscati, Christian Wallmann, Markus Reiter-Haas, Dominik Kowald, Elisabeth Lex, and Markus Schedl. Integrating the act-r framework with collaborative filtering for explainable sequential music recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, 2023.
- [197] Peter Muellner, Dominik Kowald, and Elisabeth Lex. Robustness of meta matrix factorization against strict privacy constraints. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021*, pages 107–119. Springer, 2021.

- [198] Peter Muellner, Stefan Schmerda, Dieter Theiler, Stefanie Lindstaedt, and Dominik Kowald. Towards employing recommender systems for supporting data and algorithm sharing. In *DataEconomy Workshop co-located with the 18th International Conference on emerging Networking EXperiments and Technologies*, CoNext '22, 2022.
- [199] Peter Müllner, Elisabeth Lex, and Dominik Kowald. Position paper on simulating privacy dynamics in recommender systems. In *SimuRec Workshops co-located with the 15th ACM Conference on Recommender Systems: RECSYS 2021*, 2021.
- [200] Peter Müllner, Elisabeth Lex, Markus Schedl, and Dominik Kowald. Differential privacy in collaborative filtering recommender systems: A review. *Frontiers in Big Data*, 6, 2023.
- [201] Peter Müllner, Elisabeth Lex, Markus Schedl, and Dominik Kowald. Reuseknn: Neighborhood reuse for differentially private knn-based recommendations. *ACM Trans. Intell. Syst. Technol.*, 14(5), 2023.
- [202] Peter Müllner, Dominik Kowald, and Elisabeth Lex. User groups for robustness of meta matrix factorization against decreasing privacy budgets, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.4031011>, 2020.
- [203] Peter Müllner, Dominik Kowald, Markus Schedl, Christine Bauer, Evarle Zange, and Elisabeth Lex. Lfm-beyms, zenodo dataset, doi: <https://doi.org/10.5281/zenodo.3784765>, 2020.
- [204] Athanasios N Nikolakopoulos, Xia Ning, Christian Desrosiers, and George Karypis. Trust your neighbors: A comprehensive survey of neighborhood-based methods for recommender systems. *Recommender Systems Handbook*, pages 39–89, 2021.
- [205] Xia Ning and George Karypis. Slim: Sparse linear methods for top-n recommender systems. In *2011 IEEE 11th International Conference on Data Mining*, pages 497–506. IEEE, 2011.
- [206] Eirini Ntoutsi, Pavlos Fafalios, Ujwal Gadiraju, Vasileios Iosifidis, Wolfgang Nejdl, Maria-Ester Vidal, Salvatore Ruggieri, Franco Turini, Symeon Papadopoulos, Emmanouil Krasanakis, et al. Bias in data-driven artificial intelligence systems — an introductory survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3):e1356, 2020.
- [207] Ingrid Nunes and Dietmar Jannach. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction*, 27:393–444, 2017.
- [208] John O'Donovan and Barry Smyth. Trust in recommender systems. In *Proceedings of the 10th International Conference on Intelligent User Interfaces*, pages 167–174, 2005.

- [209] Özlem Özgöbek, Jon Atle Gulla, and R Cenk Erdur. A survey on challenges and methods in news recommendation. In *International Conference on Web Information Systems and Technologies*, volume 2, pages 278–285. SCITEPRESS, 2014.
- [210] Alexandros Paramythis, Stephan Weibelzahl, and Judith Masthoff. Layered evaluation of interactive adaptive systems: Framework and formative methods. *User Modeling and User-Adapted Interaction*, 20:383–453, 2010.
- [211] C Whan Park and V Parker Lessig. Familiarity and its impact on consumer decision biases and heuristics. *Journal of Consumer Research*, 8(2):223–230, 1981.
- [212] Yoon-Joo Park and Alexander Tuzhilin. The long tail of recommender systems and how to leverage it. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, pages 11–18, 2008.
- [213] Denis Parra, Peter Brusilovsky, and Christoph Trattner. See what you want to see: Visual user-driven approach for hybrid recommendation. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, pages 235–240, 2014.
- [214] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 701–710, 2014.
- [215] Pearl Pu, Li Chen, and Rong Hu. A user-centric evaluation framework for recommender systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, pages 157–164, 2011.
- [216] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. Sequence-aware recommender systems. *ACM Computing Surveys (CSUR)*, 51(4):1–36, 2018.
- [217] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 130–137, 2017.
- [218] Naren Ramakrishnan, Benjamin J Keller, Batul J Mirza, Ananth Y Grama, and George Karypis. When being weak is brave: Privacy in recommender systems. *IEEE Internet Computing*, pages 54–62, 2001.
- [219] Matěj Račinský. Myanimelist dataset, kaggle dataset, doi: <https://doi.org/10.34740/kaggle/dsv/45582>, 2018.
- [220] Markus Reiter-Haas, Emilia Parada-Cabaleiro, Markus Schedl, Elham Motamed, Marko Tkalcic, and Elisabeth Lex. Predicting music relistening behavior using the act-r framework. In *Proceedings of the 15th ACM Conference on Recommender Systems*, pages 702–707, 2021.

- [221] Markus Reiter-Haas, David Wittenbrink, and Emanuel Lacic. On the heterogeneous information needs in the job domain: A unified platform for student career. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pages 573–574, 2020.
- [222] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 452–461, 2009.
- [223] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pages 175–186, 1994.
- [224] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [225] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In *Recommender Systems Handbook*, pages 1–35. Springer, 2010.
- [226] Elaine Rich. User modeling via stereotypes. *Cognitive Science*, 3(4):329–354, 1979.
- [227] Mark Sanderson et al. Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375, 2010.
- [228] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, pages 285–295, 2001.
- [229] Markus Schedl. The lfm-1b dataset for music retrieval and recommendation. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, pages 103–110, 2016.
- [230] Markus Schedl, Christine Bauer, Wolfgang Reisinger, Dominik Kowald, and Elisabeth Lex. Listener modeling and context-aware music recommendation based on country archetypes. *Frontiers in Artificial Intelligence*, 3, 2021.
- [231] Markus Schedl, Stefan Brandl, Oleg Lesota, Emilia Parada-Cabaleiro, David Penz, and Navid Rekabsaz. Lfm-2b: A dataset of enriched music listening events for recommender systems research and fairness analysis. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, pages 337–341, 2022.
- [232] Markus Schedl and Bruce Ferwerda. Large-scale analysis of group-specific music genre taste from collaborative tags. In *2017 IEEE International Symposium on Multimedia (ISM)*, pages 479–482. IEEE, 2017.

- [233] Markus Schedl, Peter Knees, Brian McFee, Dmitry Bogdanov, and Marius Kaminskas. Music recommender systems. *Recommender Systems Handbook*, pages 453–492, 2015.
- [234] Markus Schedl, Hamed Zamani, Ching-Wei Chen, Yashar Deldjoo, and Mehdi Elahi. Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, 7:95–116, 2018.
- [235] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260, 2002.
- [236] Sebastian Scher, Bernhard Geiger, Simone Kopeinik, Andreas Trügler, and Dominik Kowald. A conceptual model for leaving the data-centric approach in machine learning. *arXiv preprint arXiv:2302.03361*, 2023.
- [237] Sebastian Scher, Simone Kopeinik, Andreas Trügler, and Dominik Kowald. Modelling the long-term fairness dynamics of data-driven targeted help on job seekers. *Scientific Reports*, 13(1):1727, 2023.
- [238] Paul Seitlinger, Dominik Kowald, Simone Kopeinik, Ilire Hasani-Mavriqi, Elisabeth Lex, and Tobias Ley. Attention please! a hybrid resource recommender mimicking attention-interpretation dynamics. In *Proceedings of the 24th International Conference on World Wide Web*, pages 339–345, 2015.
- [239] Paul Seitlinger, Dominik Kowald, Christoph Trattner, and Tobias Ley. Recommending tags with a model of human categorization. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, pages 2381–2386, 2013.
- [240] Paul Seitlinger, Tobias Ley, Dominik Kowald, Dieter Theiler, Ilire Hasani-Mavriqi, Sebastian Dennerlein, Elisabeth Lex, and Dietrich Albert. Balancing the fluency-consistency tradeoff in collaborative information search with a recommender approach. *International Journal of Human-Computer Interaction*, 34(6):557–575, 2018.
- [241] Harald Semmelrock, Simone Kopeinik, Dieter Theiler, Tony Ross-Hellauer, and Dominik Kowald. Reproducibility in machine learning-driven research. *arXiv preprint arXiv:2307.10320*, 2023.
- [242] Yue Shi, Martha Larson, and Alan Hanjalic. List-wise learning to rank with matrix factorization for collaborative filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, pages 269–272, 2010.
- [243] Yue Shi, Martha Larson, and Alan Hanjalic. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)*, 47(1):1–45, 2014.

- [244] Rashmi Sinha and Kirsten Swearingen. The role of transparency in recommender systems. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pages 830–831, 2002.
- [245] Manel Slokom, Alan Hanjalic, and Martha Larson. Towards user-oriented privacy for recommender system data: A personalization-based approach to gender obfuscation for user profiles. *Information Processing & Management*, 58(6):102722, 2021.
- [246] Nasim Sonboli, Robin Burke, Michael Ekstrand, and Rishabh Mehrotra. The multisided complexity of fairness in recommender systems. *AI magazine*, 43(2):164–176, 2022.
- [247] Harald Steck. Calibrated recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, page 154–162, New York, NY, USA, 2018. Association for Computing Machinery.
- [248] Martin Stettinger, Alexander Felfernig, Gerhard Leitner, and Stefan Reiterer. Counteracting anchoring effects in group decision making. In *23rd International Conference on User Modeling, Adaptation and Personalization (UMAP 2015)*, pages 118–130. Springer, 2015.
- [249] Martin Stettinger, Alexander Felfernig, Gerhard Leitner, Stefan Reiterer, and Michael Jeran. Counteracting serial position effects in the choicla group decision support environment. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 148–157, 2015.
- [250] Zhu Sun, Jie Yang, Jie Zhang, Alessandro Bozzon, Long-Kai Huang, and Chi Xu. Recurrent knowledge graph embedding for effective recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 297–305, 2018.
- [251] Özge Sürer, Robin Burke, and Edward C Malthouse. Multistakeholder recommendation with provider constraints. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 54–62, 2018.
- [252] Nava Tintarev, Matt Dennis, and Judith Masthoff. Adapting recommendation diversity to openness to experience: a study of human behaviour. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 190–202. Springer, 2013.
- [253] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *2007 IEEE 23rd International Conference on Data Engineering Workshop*, pages 801–810. IEEE, 2007.
- [254] Nava Tintarev and Judith Masthoff. Designing and evaluating explanations for recommender systems. In *Recommender Systems Handbook*, pages 479–510. Springer, 2010.

- [255] Nava Tintarev and Judith Masthoff. Explaining recommendations: Design and evaluation. In *Recommender Systems Handbook*, pages 353–382. Springer, 2015.
- [256] Marko Tkalcic and Li Chen. Personality and recommender systems. In *Recommender Systems Handbook*, pages 715–739. Springer, 2015.
- [257] Thi Ngoc Trang Tran, Alexander Felfernig, and Nava Tintarev. Humanized recommender systems: State-of-the-art and research issues. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 11(2):1–41, 2021.
- [258] Endel Tulving. What is episodic memory? *Current Directions in Psychological Science*, 2(3):67–70, 1993.
- [259] Sarah Underwood. Regulation of ai remains elusive, communications of the acm news, url: <https://cacm.acm.org/news/248474-regulation-of-ai-remains-elusive/>, 2020. Accessed on November 22nd, 2023.
- [260] André Calero Valdez and Martina Ziefle. The users’ perspective on the privacy-utility trade-offs in health recommender systems. *International Journal of Human-Computer Studies*, 121:108–121, 2019.
- [261] Joaquin Vanschoren. Meta-learning. *Automated Machine Learning: Methods, Systems, Challenges*, pages 35–61, 2019.
- [262] Saúl Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, pages 109–116, 2011.
- [263] Sahil Verma and Julia Rubin. Fairness definitions explained. In *Proceedings of the International Workshop on Software Fairness*, pages 1–7, 2018.
- [264] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1096–1103, 2008.
- [265] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1235–1244, 2015.
- [266] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. A survey on the fairness of recommender systems. *ACM Transactions on Information Systems*, 41(3):1–43, 2023.
- [267] Stanley L Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965.

- [268] Udi Weinsberg, Smriti Bhagat, Stratis Ioannidis, and Nina Taft. Blurme: Inferring and obfuscating user gender based on ratings. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, pages 195–202, 2012.
- [269] Cort J Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79–82, 2005.
- [270] Raymond E. Wright. Logistic regression. *Reading and Understanding Multivariate Statistics*, pages 217–244, 1995.
- [271] Ke Yang and Julia Stoyanovich. Measuring fairness in ranked outputs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, pages 1–6, 2017.
- [272] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.
- [273] Eva Zangerle and Christine Bauer. Evaluating recommender systems: Survey and framework. *ACM Computing Surveys*, 55(8):1–38, 2022.
- [274] Guijuan Zhang, Yang Liu, and Xiaoning Jin. A survey of autoencoder-based recommender systems. *Frontiers of Computer Science*, 14:430–450, 2020.
- [275] Mingwu Zhang, Yu Chen, and Jingqiang Lin. A privacy-preserving optimization of neighbourhood-based recommendation for medical-aided diagnosis and treatment. *IEEE Internet of Things Journal*, 2021.
- [276] Minxing Zhang, Zhaochun Ren, Zihan Wang, Pengjie Ren, Zhunmin Chen, Pengfei Hu, and Yang Zhang. Membership inference attacks against recommender systems. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 864–879, 2021.
- [277] Shijie Zhang, Wei Yuan, and Hongzhi Yin. Comprehensive privacy analysis on federated recommender system against attribute inference attacks. *IEEE Transactions on Knowledge and Data Engineering*, 2023.
- [278] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1):1–38, 2019.
- [279] Yuan Cao Zhang, Diarmuid Ó Séaghdha, Daniele Quercia, and Tamas Jambor. Auralist: Introducing serendipity into music recommendation. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, pages 13–22, 2012.
- [280] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web*, pages 22–32, 2005.

Appendices

Appendix A

Own Contributions to Main Publications

This chapter describes my own contributions to the 17 main publications of this cumulative habilitation. All of these publications were created in a joint effort with my co-authors, and I would like to thank them again here for the great collaborations that made these publications possible.

Furthermore, the habilitation guidelines of Graz University of Technology require that own contributions to papers with co-authors are highlighted. I do this in Table A.1 by stating my contributions to the publications below each paper reference in the table. Wherever possible, the stated contributions are in line with the author contribution sections of the given journal papers.

Table A.1: Description of own contributions to the main publications selected by the author of the habilitation.

No.	Publication
Transparency and Cognitive Models in Recommender Systems	
P1	Seitlinger, P., Ley, T., Kowald, D. , Theiler, D., Hasani-Mavriqi, I., Dennerlein, S., Lex, E., Albert, D. (2018). Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach. <i>International Journal of Human-Computer Interaction</i> , 34:6, pp. 557-575. DOI: https://doi.org/10.1080/10447318.2017.1379240
<p>I contributed to the research idea of this paper and developed large parts of the bookmarking interface, which was used to conduct the study. I also developed the tag recommendation algorithms (<i>MostPopular</i> and <i>SoMe</i>), integrated them into the bookmarking interface, as well as contributed to the technical user study setup, data collection procedure, and the evaluation of the recommendation results. Additionally, I contributed to writing the paper throughout all iterations of writing. Apart from that, I was the first author of a short version of this publication, which I presented in the poster track of TheWebConf'2018.</p>	

-
- P2** Lex, E.*, **Kowald, D.***, Schedl, M. (2020). Modeling Popularity and Temporal Drift of Music Genre Preferences. *Transactions of the International Society for Music Information Retrieval*, 3:1, pp. 17-30. (*equal contribution) DOI: <https://doi.org/10.5334/tismir.39>

I shared the first authorship of the paper with Elisabeth Lex. Together, we created the research idea, methodology, and main text of this paper. Apart from that, I created the Last.fm dataset sample used in the paper, identified the different user groups in the dataset, developed the cognitive-inspired recommendation algorithms, and evaluated them using the *TagRec* framework, for which I am the main developer. I also created all tables and figures presented in the paper.

-
- P3** **Kowald, D.***, Lex, E.*, Schedl, M. (2020). Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations. In *4th Workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory (HUMANIZE @ ACM IUI'2020)*. (*equal contribution) DOI: <https://doi.org/10.48550/arXiv.2003.10699>

As in the case of **P2**, I shared the first authorship of this paper with Elisabeth Lex, and together, we created the research idea, methodology, and main text of the paper. In addition, I developed and evaluated the semantic context component of the activation equation of the cognitive model ACT-R, and integrated it into the *TagRec* framework. I discussed the difference between the full activation equation and base-level learning equation of ACT-R, and created all tables and figures.

-
- P4** Lex, E., **Kowald, D.**, Seitlinger, P., Tran, T., Felfernig, A., Schedl, M. (2021). Psychology-informed Recommender Systems. *Foundations and Trends in Information Retrieval*, 15:2, pp. 134–242. DOI: <https://doi.org/10.1561/1500000090>

I contributed to the general idea and the survey method of this paper. I also contributed to the sections on cognitive-inspired recommender systems and cognitive models of attention, to all discussion subsections in the paper, the formalization of activation process in human memory, and created the schematic illustration of the ACT-R architecture. Finally, I contributed to writing the paper throughout all iterations of writing, and supported in identifying potential avenues for future research, as well as discussing how cognitive models contribute to transparency aspects.

-
- P5** Moscati, M., Wallmann, C., Reiter-Haas, M., **Kowald, D.**, Lex, E., Schedl, M. (2023). Integrating the ACT-R Framework and Collaborative Filtering for Explainable Sequential Music Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems (RecSys'2023)*, pp. 840–847. DOI: <https://doi.org/10.1145/3604915.3608838>

I contributed to the description of the method and experimental setup, and to the description and interpretation of the results, as well as to paper writing during all iterations. Specifically, I contributed to formalizing the components of the recommendation approach based on ACT-R, and to interpreting the weights of the ACT-R components towards providing transparent and explainable recommendations.

Privacy and Limited Preference Information in Recommender Systems

- [P6] Duricic, T., Lacic, E., **Kowald, D.**, Lex, E. (2018). Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys'2018)*, pp. 446–450. DOI: <https://doi.org/10.1145/3240323.3240404>

I contributed to formalizing the approach based on Katz similarity, and defining the user cold-start experimental setup used to evaluate the trust-based recommendations. Additionally, I was involved in the interpretation of the evaluation results, and all iterations of paper writing. This paper is part of Tomislav Duricic's (first author) ongoing Ph.D. thesis, for which I am co-supervisor together with Elisabeth Lex.

- [P7] Lacic, E., Reiter-Haas, M., **Kowald, D.**, Dareddy, M., Cho, J., Lex, E. (2020). Using Autoencoders for Session-based Job Recommendations. *User Modeling and User-Adapted Interaction*, 30, pp. 617–658. DOI: <https://doi.org/10.1007/s11257-020-09269-1>

I contributed to empirical research, the experimental setup, description of the session-based recommendation approach based on limited preference information of the users, definition of the autoencoder-based system architecture, interpretation and discussion of results, and paper writing in all iterations. Specifically, I contributed to the formal definition and implementation of the beyond-accuracy metrics system-based novelty and session-based novelty.

- [P8] Muellner, P., **Kowald, D.**, Lex, E. (2021). Robustness of Meta Matrix Factorization Against Strict Privacy Constraints. In *Proceedings of the 43rd European Conference on Information Retrieval (ECIR'2021)*, pp. 107-119. DOI: https://doi.org/10.1007/978-3-030-72240-1_8

I contributed to the original research idea of this paper, to the methodology of the reproducibility and privacy-focused studies, description and interpretation of the results, and paper writing in all iterations. I especially supported in defining the experiments to study the users' privacy constraints. This paper is part of Peter Muellner's (first author) ongoing Ph.D. thesis, for which I am co-supervisor together with Elisabeth Lex. Apart from that, I was the last author of a short version of this publication, which was presented at the Responsible AI Forum 2021.

-
- P9** Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). ReuseKNN: Neighborhood Reuse for Differentially-Private KNN-Based Recommendations. *ACM Transactions on Intelligent Systems and Technology*, 14:5, pp. 1-29. DOI: <https://doi.org/10.1145/3608481>

As last author, I contributed to the original research idea, finding and describing related work, defining the problem setting, formalizing the approach and the evaluation settings, and interpreting the evaluation results. I also contributed to discussing the trade-off between privacy and accuracy. Additionally, I contributed to paper writing in all iterations. This paper is part of Peter Muellner's (first author) ongoing Ph.D. thesis, for which I am co-supervisor together with Elisabeth Lex.

-
- P10** Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). Differential Privacy in Collaborative Filtering Recommender Systems: A Review. *Frontiers in Big Data*, 6:1249997, pp. 1-7. DOI: <https://doi.org/10.3389/fdata.2023.1249997>

As last and corresponding author of this article, I contributed to the original idea, conceptualization, writing process throughout all iterations, and supervision of the review methodology and the paper writing process. Additionally, I contributed to the categorization of the 26 publications reviewed in this article, and to the identification of open research questions in the field of differentially private recommender systems. This article is part of Peter Muellner's (first author) ongoing Ph.D. thesis, for which I am co-supervisor together with Elisabeth Lex.

Fairness and Popularity Bias in Recommender Systems

-
- P11** **Kowald, D.**, Schedl, M., Lex, E. (2020). The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In *Proceedings of the 42nd European Conference on Information Retrieval (ECIR'2020)*, pp. 35-42. DOI: https://doi.org/10.1007/978-3-030-45442-5_5

As first and corresponding author of this paper, I contributed to the research idea, created the first full draft of the paper, created the Last.fm dataset sample, implemented the recommendation algorithms and evaluation methods, conducted the experiments, and described and interpreted the results. The source-code for this publication started my *Fair-RecSys GitHub* repository, which contains Python scripts for studying fairness and popularity bias in recommender systems. I also presented the paper in the reproducibility track of the *European Conference on Information Retrieval (ECIR'2020)*.

-
- P12** **Kowald, D.**, Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E. (2021). Support the Underground: Characteristics of Beyond-Mainstream Music Listeners. *EPJ Data Science*, 10:14. DOI: <https://doi.org/10.1140/epjds/s13688-021-00268-9>

As first and corresponding author of this publication, I contributed to the original idea, the collection of related work, identification of beyond-mainstream users in the Last.fm dataset, data analysis methods, and description and interpretation of results, as well as large parts of the paper writing process during all iterations. I also contributed to establish the connection between the recommendation accuracy results of the subgroups and openness patterns of these subgroups. This paper was part of Peter Muellner's (second author) Master's thesis, for which I was co-supervisor together with Elisabeth Lex. I was also involved in interviews discussing the findings of this paper for several news outlets (e.g., Rolling Stone Italy or BioMed Central).

-
- P13** Lesota, O., Melchiorre, A., Rekabsaz, N., Brandl, S., **Kowald, D.**, Lex, E., Schedl, M. (2021). Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected? In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'2021)*, pp. 601-606. DOI: <https://doi.org/10.1145/3460231.3478843>

I contributed to the conceptualization, finding and description of related work, methodology for measuring popularity bias across genders, investigation and interpretation of the results, and paper writing in all iterations. Specifically, I contributed to defining delta metrics for measuring popularity bias based on the delta group average popularity metric, which was proposed for music recommendations in **P11**.

-
- P14** **Kowald, D.**, Lacic, E. (2022). Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2022)*. Communications in Computer and Information Science, vol. 1610, pp. 1-11. DOI: https://doi.org/10.1007/978-3-031-09316-6_1

As first and corresponding author of this paper, I contributed to the research idea, created the first full draft of the paper, created the dataset samples and user group divisions, implemented the recommendation algorithms and evaluation methods using my *FairRecSys GitHub* repository, conducted the experiments, and described and interpreted the results. I also presented the paper at the *European Conference on Information Retrieval (ECIR'2022)*. Together with Emanuel Lacic, I was awarded with the *Mind-the-Gap Gender and Diversity* award of Graz University of Technology for this paper.

-
- P15** Lacic, E., Fadljevic, L., Weissenboeck, F., Lindstaedt, S., **Kowald, D.** (2022). What Drives Readership? An Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations. In *Proceedings of the 44th European Conference on Information Retrieval (ECIR'2022)*, pp. 172-179. DOI: https://doi.org/10.1007/978-3-030-99739-7_20

As last and corresponding author of this paper, I contributed to the original research idea, the design of the content-based news article recommendation algorithm, the definition of the research questions and experimental setup, the design of the online user study, the choice of suitable evaluation metrics, and the description and interpretation of the results. I did the main communications with representatives of the news platform *DiePresse*, and created a first full draft of the paper together with the first author, Emanuel Lacic. Together, we presented the paper at the *European Conference on Information Retrieval (ECIR'2022)*.

-
- P16 **Kowald, D.***, Mayr, G.*., Schedl, M., Lex, E. (2023). A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2023)*. Communications in Computer and Information Science, vol. 1840, pp. 1-16. (*equal contribution) DOI: https://doi.org/10.1007/978-3-031-37249-0_1

As first and corresponding author of this publication, I contributed to the original idea, the methodology, the creation of the dataset samples, interpretation of results, and paper writing in all iterations. Specifically, I extracted the genre information of the three datasets, and assigned the genres to the corresponding items using my *FairRecSys GitHub* repository. This paper was part of Gregor Mayr's (co-first author) Bachelor's thesis and Master's project, for which I was co-supervisor together with Elisabeth Lex. I also presented the paper at the *European Conference on Information Retrieval (ECIR'2023)*.

-
- P17 Scher, S., Kopeinik, S., Truegler, A., **Kowald, D.** (2023). Long-Term Dynamics of Fairness: Understanding the Impact of Data-Driven Targeted Help on Job Seekers. *Nature Scientific Reports*, 13:1727. DOI: <https://doi.org/10.1038/s41598-023-28874-9>

As last author, I contributed to the analysis of the data, the discussion and interpretation of the empirical results, and writing of the manuscript in all iterations of the writing process. Additionally, I contributed to formalizing and describing the trade-off between the different long-term fairness goals, and to relating them to the trade-off between individual and group fairness. Together with the first two authors of this publication, I set up Master thesis topics to transfer the methodology of this study to the area of fair recommender systems.

As described in this table, I contributed substantially to all 17 publications, and for 10 of these publications I am also either first or last author. In the following, the full texts of the papers are given. I use the published journal and conference formats for all papers, except for **P4**, where I altered the formatting slightly due to copyright restrictions of the journal's publisher.

Appendix B

Full Texts of Main Publications

B.1 Transparency and Cognitive Models in Recommender Systems

P1 Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach (2018)

Transparency and Cognitive Models in Recommender Systems

[P1] Seitlinger, P., Ley, T., **Kowald, D.**, Theiler, D., Hasani-Mavriqi, I., Dennerlein, S., Lex, E., Albert, D. (2018). Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach. *International Journal of Human-Computer Interaction*, 34:6, pp. 557-575.
DOI: <https://doi.org/10.1080/10447318.2017.1379240>

Balancing the Fluency-Consistency Tradeoff in Collaborative Information Search with a Recommender Approach

Paul Seitlinger^a, Tobias Ley^a, Dominik Kowald^b, Dieter Theiler^b, Ilire Hasani-Mavriqi^{b,c}, Sebastian Dennerlein^b, Elisabeth Lex^b, and Dietrich Albert^{b,c,d}

^aSchool of Educational Sciences, Tallinn University, Tallinn, Estonia; ^bInstitute of Interactive Systems and Data Science, Graz University of Technology, Graz, Austria; ^cKnow-Center GmbH, Research Center for Data-Driven Business & Big Data Analytics, Graz, Austria; ^dInstitute of Psychology, University of Graz, Graz, Austria

ABSTRACT

Creative group work can be supported by collaborative search and annotation of Web resources. In this setting, it is important to help individuals both stay fluent in generating ideas of what to search next (i.e., maintain ideational fluency) and stay consistent in annotating resources (i.e., maintain organization). Based on a model of human memory, we hypothesize that sharing search results with other users, such as through bookmarks and social tags, prompts search processes in memory, which increase ideational fluency, but decrease the consistency of annotations, e.g., the reuse of tags for topically similar resources. To balance this tradeoff, we suggest the tag recommender SoMe, which is designed to simulate search of memory from user-specific tag-topic associations. An experimental field study ($N = 18$) in a workplace context finds evidence of the expected tradeoff and an advantage of SoMe over a conventional recommender in the collaborative setting. We conclude that sharing search results supports group creativity by increasing the ideational fluency, and that SoMe helps balancing the evidenced fluency-consistency tradeoff.

KEYWORDS

Ideational fluency; tagging consistency; exploration-exploitation tradeoff; collaborative search; tag recommender; reflective search framework

1. Introduction

Imagine a team of people in product development or organizational design with diverse backgrounds that have been given the task to deliver new kinds of products or solutions for designing new workspaces. They may try to research recent trends in workspace design to come up with innovative ideas. In today's work, such teamwork has become commonplace to deal with complex problems or for finding innovative solutions. In such a setting, it is necessary that the persons involved learn from one another, draw on their creativity, overcome groupthink (Janis, 1972; Page, 2007) and to come up with innovative product ideas (Paulus & Brown, 2007).

If the group uses information technology to do research and communicate, then this constitutes a networked search of solutions (e.g., Lazer & Bernstein, 2012), where each member receives help from human and non-human sources and contributes to a collective attempt to connect all those sources to a creative solution that is novel and useful. Digital curation, i.e., collaborating on the search and organization of problem-related sources (e.g., articles, videos, etc.) in social Web environments (e.g., Kerne, Smith, Koh, Choi, & Graeber, 2008; Kerne et al., 2014; Linder, Snodgrass, & Kerne, 2014), can be key in networked search. By raising one's awareness of others' contributions (e.g., collected sources) and reflections upon them (e.g., annotations), digital curation helps to mutually stimulate ideas "to increase one's potential for realizing

creativity" (Linder et al., 2014, p. 2411). In the sense of Sarmiento and Stahl (2008), digital curation supports the social dimension of creativity because it facilitates the building and maintenance of a shared problem space (e.g., of emerging relations between Web resources mediated by annotations) and bridging across different individuals' complementary ideas.

Lazer and Bernstein (2012) and Lazer and Friedman (2007) reported a number of theoretical and simulation-based studies in the tradition of networked search suggesting that effective search needs to balance divergent and convergent search processes. Divergent processes give each individual agent enough room for experimentation (e.g., autonomous exploration of information). From time to time, this needs to be balanced by convergent processes that allow exploiting and aligning each other's approaches toward a solution. The latter is usually accomplished by providing appropriate communication structures through which agents align their understanding.

Different research perspectives on social tagging (e.g., Fu, Kannampallil, Kang, & He, 2010; Lorince & Todd, 2016; Nelson et al., 2009; Pirolli & Kairam, 2012; Schweiger, Oeberst, & Cress, 2014) suggest that the use of tag-based annotations in digital curation could support such balancing. Tags are freely chosen keywords with which users describe resources on the Web and which may be visible to others. On the one hand, as social tags reveal other members' thoughts,

they trigger cognitive conflicts and inspire new ideas during individual experimentation (e.g., Schweiger et al., 2014). On the other hand, given sufficient consistency in applying certain tags for reoccurring topics, they support tag-based sharing of collected resources and facilitate an exploitation of own and others' search results (e.g. Fu et al., 2010; Lorince & Todd, 2016; Nelson et al., 2009; Pirolli & Kairam, 2012).

According to networked search (e.g., Lazer & Bernstein, 2012), a balanced view on the divergence–convergence continuum should lead to effective designs of digital curation environments. However, in social media and particularly social tagging studies, questions around the convergent pole have dominated the discourse and its agenda (e.g., Golder & Huberman, 2006). Probably because the lack of central control (e.g., standardized vocabularies) can lead to the “vocabulary problem” (divergent wording when tagging the very same object; Furnas, Landauer, Gomez, & Dumais, 1987), an endeavor of investigating and supporting consistency has come to the fore, for example by studying “semantic stabilization” (Wagner, Singer, Strohmaier, & Huberman, 2014). This focus on the convergent pole is also reflected by a large body of literature on the development of automatic tag recommendation mechanisms (TRM) (Dellschaft & Staab, 2012; Font, Serrà, & Serra, 2015; Jäschke, Marinho, Hotho, Schmidt-Thieme, & Stumme, 2007). TRM are services that encourage a convergent tag use and hence, alleviate the vocabulary problem by suggesting tags already applied in the past by other users. An example for a very simple strategy is represented by “most popular” recommenders which assume that what has been applied by many in the past is a good predictor for future assignments. Despite their simplicity, Most Popular Tag (MPT) recommenders work surprisingly well in predicting tag reuse in offline studies (Jäschke et al., 2007; Kowald et al., 2014).

In this article, we address the question of how to balance both processes in tag-based digital curation, i.e., how to increase exploration (divergent thinking of new search topics) against the backdrop of a sufficient level of tagging consistency in support of exploitation (making use of other persons' tags and associated search results). We investigate this question in a scenario where creative solutions are particularly important, namely in a work-integrated information search, and where a strong focus on convergent processes may especially be detrimental because persons are likely to share a common background and information goal, so that divergent processes need to be stressed to allow for creative solutions.

Rather than focusing on social imitation as many of the previous works have done in the area of tag-based curation, we approach this problem by drawing on the framework of “reflective search” (Seitlinger & Ley, 2016). The reflective search framework regards human web interaction as an iterative search of human memory shaped by past and present learning episodes. In our previous article, we have especially focused on convergent processes by looking at stabilizing tag vocabulary. In the present article, we draw our attention to the divergent pole of the exploration–exploitation continuum by considering effects of networked search on ideational fluency, a concept from the creative cognition literature (e.g., Benedek & Neubauer, 2013). In the present context, it

describes how easily and continuously diverse ideas can be accessed from memory during information-based ideation (Kerne et al., 2008, 2014), i.e., when thinking about search topics to be explored in future queries. Referring to previous work on cognitive effects of social tags on mental structures (e.g., categories and associations; e.g., Fu & Dong, 2012; Seitlinger & Ley, 2012; Seitlinger, Ley, & Albert, 2015), we anticipate a tradeoff between fluency and consistency: when users of a digital curation environment perceive others' tags, these tags leave episodic memory traces (Seitlinger & Ley, 2012; Seitlinger et al., 2015), strengthening previously weak associations to a search topic, in case these traces represent new ideas. Considering research in creative cognition (e.g., Smith, Ward, & Finke, 1995; Ward, 2007), this tag-based cognitive effect should reduce the dominance of pre-existing stereotyped associations and give rise to a broader (mental) fan of equally available ideas around a topic. This is also called a flatter hierarchy of associative strengths (Benedek & Neubauer, 2013; Mednick, 1962). While such a mental organization allows for a steadier stream of ideas, i.e., more fluency, the increased availability of several responses to a certain topic should simultaneously decrease tagging consistency.

The first research question of this work is *whether the anticipated fluency-consistency tradeoff can be evidenced when people perform tag-based search and curation collaboratively (seeing each other's tags and search results) in contrast to when they perform this search individually*. In particular, we seek to demonstrate the flattening impact of social cues (e.g., tagged bookmarks) on people's associative hierarchies by the manifestation of the tradeoff (decreased consistency and increased fluency) in a realistic information search scenario at the workplace. We let persons (research staff) bookmark and tag Web resources on a given topic ('redesigning workspaces to move people') both under an individual and collaborative search condition. Under both conditions, we then examine each person's tagging consistency (extent of choosing similar tags for topically similar bookmarks) and ideational fluency (stream of responses in a free association task on a set of subtopics, such as 'interior design', 'inspiration sources', etc.) and expect the experimental variation (individual vs. collaborative) to lead to increases in fluency being accompanied by decreases in consistency, and vice versa.

In the light of the previously mentioned studies of Lazer and Bernstein (2012) and Lazer and Friedman (2007) on the tradeoff between divergent and convergent search processes in effective problem solving in groups, this research question is not novel. Moreover, in this context, studies on creative group cognition (e.g., Kohn, Paulus, & Choi, 2011; Nijstad & Stroebe, 2006) should also be mentioned, which have already found evidence that mutual stimulation in collaborative brainstorming settings can have positive effects on individual fluency scores. Therefore, investigating our first research question is primarily of an incremental nature and asks whether existing evidence gathered under controlled experimental conditions generalizes to more natural conditions of an information search at the workplace. Especially, from an applied perspective, such validation appears important to us because it would imply that designing social systems for

information search can be directed either toward convergence (i.e., tagging consistency), or toward divergence (i.e., ideational fluency). The former would aim for aligning the vocabulary (e.g., Font et al., 2015), while the latter for stimulating creation (e.g., Candy & Hewett, 2008), and it would seem difficult to balance the two complementary processes. Considering recent discussions around filter bubbles in Web environments (e.g., Pariser, 2011; Sunstein, 2001), we consider such research to contribute to a more balanced view on interaction in the social web.

As a second aim of this article, we therefore address this design challenge by introducing a tag recommendation mechanism (TRM) that compensates for the hypothesized downside of decreased consistency by pushing the reuse of tags for certain topics, even if associative hierarchies are flat. To design such a TRM, we draw on formalisms of memory psychology that help translate the reflective search framework into a computational model to be readily applied as a TRM. As summarized in Figure 1, the first step of translation, i.e., a formalization of the model, makes use of a stochastic account of human memory search (e.g., Unsworth & Engle, 2007) to specify cognitive structures (e.g., an associative hierarchy) and processes (e.g., encoding and retrieval), which are involved in reflecting on objects and underlie the tradeoff between fluency and consistency. The second translation step then instantiates the search-of-memory formalization to create a TRM, which emulates a person's associations by tracking and consolidating tag choices for particular resource topics. Based on this memory-like representation of tagging behavior, the mechanism is able to mimic a resource-triggered search of memory (SoMe) for topically relevant tags. This should confer an advantage over conventional most popular tags (MPT) approaches, if associative hierarchies are flat. *Therefore, the second research question is whether SoMe achieves high tag acceptance rates under a collaborative search where*

associative hierarchies are assumed to be flat, and, more specifically, whether the SoMe advantage (over MPT) is larger under the collaborative than individual search condition.

Summarizing, while our first research question asks for empirical evidence of the fluency-consistency tradeoff, the second points toward an effective strategy to alleviate inconsistent tagging behavior in case ideational fluency is high. We will now turn to the first research question and present an empirical study conducted to that evidence. In the article's second part, we will then tackle the second question and introduce an effective recommender approach that is applied in the same study.

2. Evidence of the fluency-consistency tradeoff in collaborative information search

2.1. Background and hypotheses

As the question of the fluency-consistency tradeoff is built upon specific assumptions on the mental organization of a person's associations (the associative hierarchy), we first provide a more formal and process-oriented interpretation of an associative hierarchy in form of a stochastic model of memory search. Based on this model, we specify the assumed effect of social cues (during a collaborative search) on an associative hierarchy and derive the hypotheses on the fluency-consistency tradeoff.

A process-oriented interpretation of an associative hierarchy

Our starting point of formalizing the associative hierarchy is the reflective search framework (Seitlinger & Ley, 2016). We assume that reflections upon objects (e.g., when tagging topics of an article, or generating ideas to approach a problem, etc.) are accompanied by a search of secondary memory. Referring

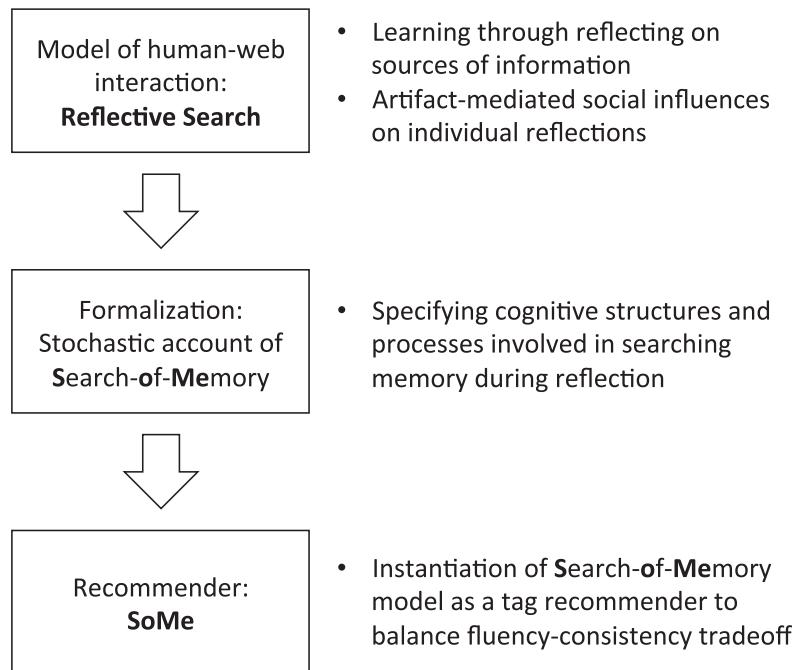


Figure 1. Design process translating reflective search model into a service that balances the fluency-consistency tradeoff.

to long-standing research on human memory search (for a review see Davelaar & Raaijmakers, 2012), this process is triggered by an environmental cue, such as a problem to be solved or an article to be tagged. This cue is assumed to activate a mental search set S (see Figure 2), a reservoir of associations, from which a number of N targets (e.g., problem-relevant ideas, or topic-related aspects; schematized by black filled dots) can be sampled (i.e., brought to mind) with a particular search rate λ .

The temporal dynamics of memory search are driven by an inverse relationship between N and λ (e.g., Albert, 1968; Bousfield, Sedgewick, & Cohen, 1954; Kaplan, Carvellas, & Metlay, 1969): If many ideas about a given problem or aspects of an article's topic can come to mind (large N ; right search set in Figure 2), the search rate is reduced (small λ) due to more competition between the available ideas – with “each [idea] competing against all of its peers” (Rohrer, 2002). Put differently, the mental search set is in a state of defocused activation (e.g., Dorfman, Martindale, Gassimova, & Vartanian, 2008; Martindale, 1995) that is distributed among several elements with comparatively low relative strengths. Such an activation state constitutes a flat associative hierarchy allowing for a slow but steady stream of ideas (Mednick, 1962). To the contrary, if only few ideas have strong associations to a given topic, the search set exhibits a state of focused activation shared among only few associations with high relative strengths (left search set in Figure 2). The resulting steep hierarchy becomes manifest in a fast retrieval (large λ) of only few ideas (small N).

As mentioned above, one way to reveal a person's associative hierarchy, i.e., to estimate N and λ , is to measure ideational fluency with respect to the topics of the collaborative

information search. To this end, we drew on a free association task, and a corresponding stochastic model to analyze the responses and derive the two parameters. In the free association task, a participant is exposed to a cue (e.g., the topic ‘interior design’) and asked to name as many associations (ideas) that come to mind. The triggered stream of responses is recorded in form of the cumulative number of unique ideas (ideational fluency) and then analyzed in terms of the temporal dynamics, i.e., inter-response times (in seconds; see schematic diagrams on the bottom of Figure 2). In the present study, we have used the topics of the information search (e.g., ‘interior design’, ‘augmented reality’, etc.) as cues for a Web-based free association task at the beginning of the study and after the individual as well as collaborative search condition.

In order to derive estimates of N and λ from the recorded ideational fluency, search through memory can be approximated as a random search process following a repeated sampling-with-replacement scheme (e.g., Bousfield & Sedgewick, 1944; Wixted & Rohrer, 1994; Unsworth & Engle, 2007). At the beginning, i.e., on the first sample, S includes N targets (relevant ideas) and $S - N$ non-targets. Since sampled targets are replaced, the rate of producing new associations, λ_i , decreases linearly with the number of already sampled targets i according to $\lambda_i = (N - i)\lambda$ (e.g., Albert, 1968). The consequence is an exponential decay function (Bousfield & Sedgewick, 1944) given by

$$F(t) = N(1 - e^{-\lambda t}) \quad (1)$$

where $F(t)$ represents the cumulative number of unique responses by time t . A large number of studies demonstrates that the cumulative exponential of Equation (1) can be fitted to the response protocol gathered by a free association task to

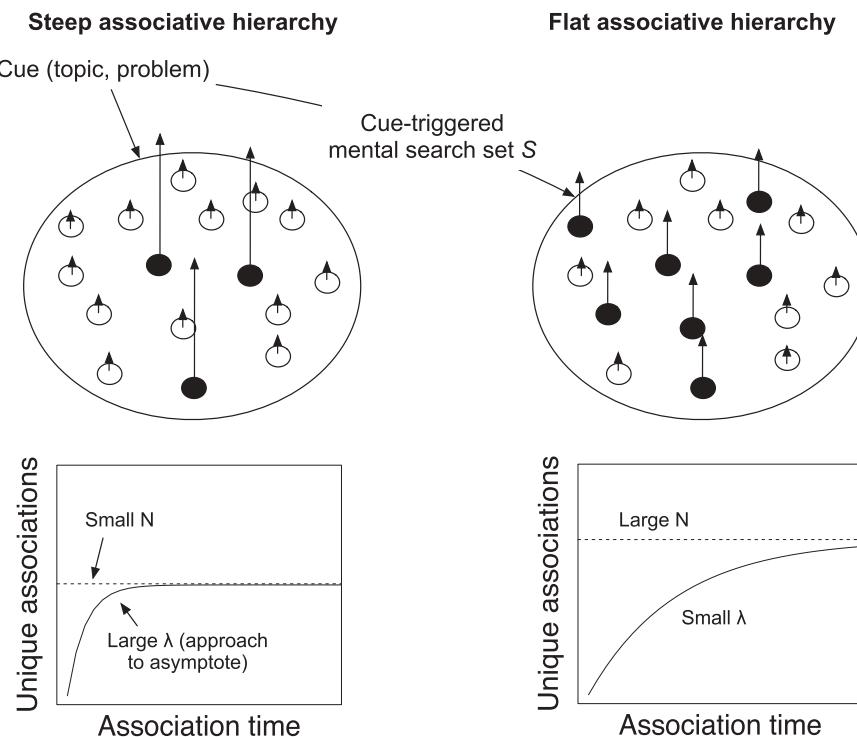


Figure 2. Associative hierarchy as a mutual dependence of N (asymptotic number of topic-related associations) and λ (rate of approach to N).

estimate N and λ (e.g., Wixted & Rohrer, 1994). In the present study, we have adopted this approach to characterize the associative hierarchies of the participants with respect to the topics explored during their information search.

Next, we make use of this stochastic search scheme to specify the effect of social cues (e.g., tagged bookmarks) on a person's associative hierarchy and to derive our hypotheses on the fluency-consistency tradeoff.

The impact of social cues on an associative hierarchy

When people collect and tag bookmarks of resources on the Web, we assume their thoughts and ideas to develop within an evolving practice that is co-created in a collective and artifact-mediated activity (e.g., Hutchins & Johnson, 2009). As a methodological consequence, we do not aim to decontextualize our unit of analysis, i.e., associative hierarchy, under laboratory conditions, but try to shed light on its relations to the natural and cultural environment and practice, in which it evolves and – at the same time – to which it contributes (e.g., Roepstorff, Niewöhner, & Beck, 2010).

To attain a holistic picture of the evolving practice, we also consider the active role of non-human actors (Latour, 2005), such as tag clouds or recommender mechanisms, which mediate the co-creation of environmental structures by distributing joint artifacts (e.g., social tags) and thus, affecting mental structures (e.g., Schweiger et al., 2014). For instance, we assume that a person's mental search set S (the set of associations activated by a given topic) is affected by joint artifacts (tags that have been introduced by other people and become visible through a tag cloud or recommendation mechanism). Evidence for the assumed long-term influence of those tags on mental associations comes from studies demonstrating perceived tags to leave robust memory traces and to affect future tag choices (e.g., Seitlinger et al., 2015). Given that some of these experienced tags convey new and interesting ideas the

person hasn't thought of before, the number N of topic-related associations (targets) should therefore increase over time within the shared bookmarking system. The assumption that tags help a person experience new ideas cannot be observed directly. However, we think the assumption is warranted, if active engagement with others' tags can be observed (e.g. through analyzing the log-file), and at the same time estimates of N are in fact larger after a collaborative than individual search. In the results section, we will be offering some more insights on whether tag-mediated mutual stimulations has likely happened in our case.

Figure 3 illustrates the expectation of tag-based influences on a person's mental search set S and contrasts a collaborative with an individual search condition. The tag clouds in the shared environment as well as the web resources others have contributed should expose a person to different perspectives on a given topic, resulting in more topic-relevant associations (dark shaped dots) within S . The consequence should be a flattening of the associative hierarchy: activation among elements in S should become defocused (larger N) and each association's relative strength should decrease and become less available (smaller λ). On the other hand, the activities in the individual bookmarking condition give rise to tag clouds resulting exclusively from an individual's contributions. In this case, the function of the tag clouds is less to propagate new than to reinforce existing associations, resulting in a smaller increase of new topic-relevant associations. This 'individualistic' interplay of environmental and mental structures mediated and reinforced by tag clouds should thus give rise to a comparatively steep associative hierarchy: a focused activation within S (small N) accompanied by a fast access to the correspondingly few associations (large λ).

As already stated, we assume the shape of an associative hierarchy (steep vs. flat) to have opposing effects on ideational fluency on the one hand, and tagging consistency on the other

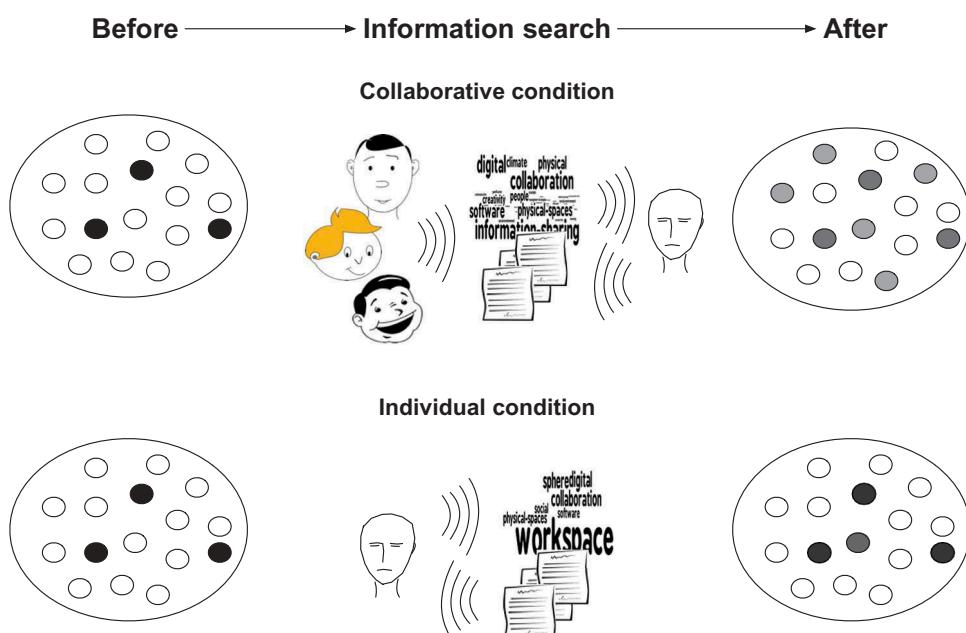


Figure 3. Artifact-mediated mutual stimulation causes associative hierarchy to be flatter after a collaborative than individual information search.

hand. This fluency-consistency tradeoff is described next as well as the two hypotheses following from it.

Testing the fluency-consistency tradeoff: Hypotheses H1.1 and H1.2

Ideational fluency was measured by means of the stream of responses in a free association task characterized by N (the asymptotic number of associations) and λ (the rate of approach to the asymptote). Based on a number of studies on retrieving from semantic memory (for a review see Wixted & Rohrer, 1994), we assumed the following relationship: The flatter the hierarchy, the larger the estimate of N and the smaller the estimate of λ (see Figure 2). In the current study, each participant performed the task three times: at the beginning of the study (baseline measurement), after an individual and after a collaborative information search (counterbalanced repeated measurement). The stimuli were eight sub-topics of the search task ('augmented reality', 'health', 'interior design', 'gamification', 'inspiration sources', 'collaboration technologies', 'personalization services', 'socializing'; for details see Section Search task and bookmarking interface) and held constant across the three points of measurement. Based on the assumption that the collaborative condition results in flatter associative hierarchies than the individual condition, *the first hypothesis was that relative to the baseline measurement, participants exhibit a stronger increase in estimates of ideational fluency (larger N and smaller λ in the free association task) after the collaborative than after the individual search (H1.1).*

Tagging consistency, the second indicator applied to characterize an associative hierarchy, was defined as the extent to which similar tags were assigned to bookmarks of semantically similar resources, i.e., resources dealing with similar topics. With respect to its relationship to the associative hierarchy, we assumed that if several associations compete for indexing a topic (flat hierarchy), the tagging behavior for resources of that topic should be more variable than if only few associations have high probabilities being retrieved (steep hierarchy). To quantify consistency, we implemented a tagging interface that prompted a person to describe each bookmark semantically (by selecting from a list of the eight search topics) and by a set of freely chosen tags. Thus, for two bookmarks x and y , we could calculate a topical similarity score S_T (normalized topic overlap of x and y) and a verbatim similarity score S_W (normalized tag overlap of x and y). Finally, we defined tagging consistency $r(S_T, S_W)$ as the correlation between S_T and S_W across all pairs of collected bookmarks separately for the individual and collaborative search condition. See Section Measures and statistical analysis for more details on the two similarity scores and calculating $r(S_T, S_W)$. Based on the assumption that the collaborative condition results in flatter associative hierarchies than the individual condition, *the second hypothesis was that estimates of $r(S_T, S_W)$ are larger under the individual than collaborative information search (H1.2).*

Summarizing, the two hypotheses H1.1 and H1.2 specify the expected fluency-consistency tradeoff that is mediated by the shape of the associative hierarchy. While a flat hierarchy favors ideational fluency, a steep one brings forward its conceptual counterpart, i.e., tagging consistency.

2.2. Method

We investigated the hypotheses H1.1 and H1.2 within the scenario of a work-integrated information search on the topic 'workspace redesign to move people', which was supported by a bookmarking system for collecting (bookmarking) and tagging topic-relevant Web resources. Our expectation was that a collaborative search condition (shared bookmarking system) on average results in a flatter associative hierarchy than an individual condition (unshared system) and that this difference manifests itself in a fluency-consistency tradeoff: a higher ideational fluency (H1.1) and a lower tagging consistency (H1.2).

Participants

The information search was performed by $N = 18$ researchers ($n = 6$ female) from different groups across different institutions, with an average age of 31.5 years ($SD = 5.5$, ranging from 23 to 46 years). The following research groups participated in the study: one cognitive science group at an Austrian technical university ($n = 6$), two groups from an Austrian research institute dealing with social computing ($n = 7$) and ubiquitous computing ($n = 3$), and one group on educational technology from an Estonian university ($n = 2$). All research groups were interdisciplinary having members on computer science, humanities, and psychology.

Particular measures were taken to ensure that persons of a shared bookmarking system could only influence each other via shared artifacts (i.e., bookmarks and tags) independent of their real geographical distance or research group membership: First, the assignment of the participants to the experimental conditions was random and did not take into account the research group membership. Second, every participant was visible in a bookmarking system only in form of a pseudonym, which was drawn randomly from a pool of popular English names (e.g., George) and did not allow inferences on her or his real name or identity. Third, participants were instructed not to discuss ongoing activities within the bookmarking system with other participating colleagues, where on average, the probability of sharing a system with a colleague from the same research group was $p = .14$. As furthermore all resources and tags had to be in English, we were confident that having two participants contribute from Estonia and the remaining sample from Austria had no significant influence on the activities going on in the bookmarking system.

Design

The independent variable, denoted 'Search Condition', differentiated between a collaborative and an individual information search. The latter took place in a separate bookmarking system only displaying each employee's own Web resources (in form of a list) as well as her/his own tags (in form of a tag cloud). Under the collaborative condition, the employees shared a social bookmarking system making available the resources and tags of all the system's members. To increase statistical power, we realized a randomized counterbalanced repeated measurement design: Every employee collected Web resources under both conditions for two weeks each, where one half of the participants switched from the individual to

the collaborative and the other half from the collaborative to the individual condition. For statistical analyses, we then merged the data of the two individual and the two collaborative study halves.

The dependent variables were (a) ideational fluency (measured by the Web-based free association task each employee performed at the beginning, after the individual, and after the collaborative search condition), and (b) tagging consistency (determined by a log file-based analysis on the correlation between the topical similarity S_T and the tag similarity S_W of an employee's bookmarks). The design included one additional independent variable (the type of Tag Recommendation Mechanism displayed), as well as one additional dependent variable (acceptance of these recommendations). We will cover this part of the design in the article's second part dealing with the tag recommendation mechanism SoMe.

Search task and bookmarking interface

The information search had the character of a simulated workplace learning scenario as the search topic 'workspace redesign to move people' was not part of an ongoing research project but defined specifically for the purpose of the study. However, to make it as realistic and motivating as possible, the topic was co-defined together with the work group leaders as a topic they expected to stimulate valuable workflow reflections and improvements. Insights gained during search were therefore discussed subsequent to the study in the context of work group meetings. While the specific search environment that was used was new to the participants, the way the search task was set up (e.g., collecting resources and sharing them in an online system) corresponded to how typically explorations of new topics were done in these institutions.

Instruction. Instructions and passwords (to enter the bookmarking system) were sent via e-mail. The instruction described the search topic, which was '*workspace (re)design to move people – improving knowledge exchange and creation in your work group*'. For the coming four weeks, the employees were asked to imagine the task of writing a state of the art for a project proposal that '*sheds light on the topic from different perspectives*'. To this end, each employee had to bookmark at least three to four Web resources (e.g., articles, videos) per week and to annotate each bookmark '*by means of predefined topics (e.g., 'inspiration sources & techniques') and freely chosen tags (e.g., 'physical_proximity', 'random_encounters', etc.)*'. Under the collaborative condition, the employee was also instructed to attend to each other's contributions (tags and shared bookmarks) to get to know different perspectives on the task.

Bookmarking interface. A Web resource was bookmarked by means of an interface displayed in Figure 4, which prompted an employee to annotate a resource by choosing one or several topics from a predefined eight-item list (number 2), and assigning tags by choosing from a list of recommendations (number 3) and/or typing in personal tags (number 4). The subset of chosen topics, denoted T , was logged for a later analysis of the employee's tagging consistency as well as to trigger the presentation of the set of

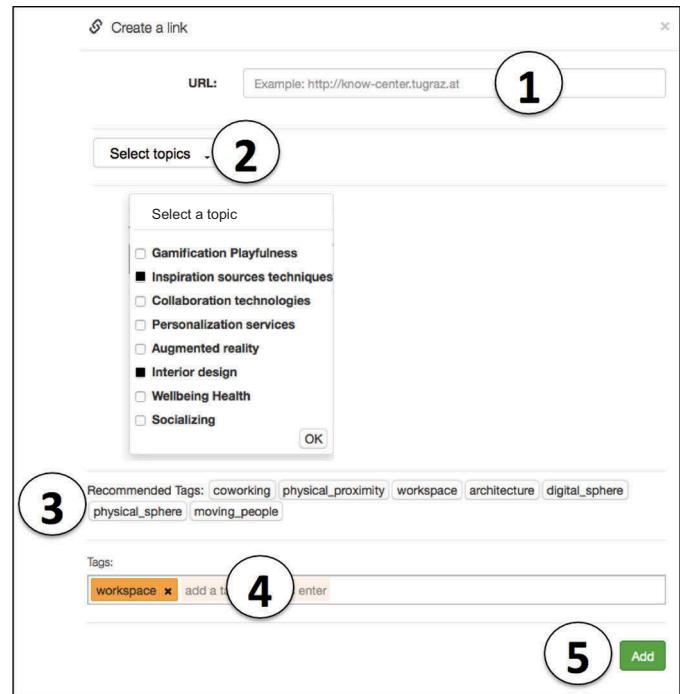


Figure 4. Bookmarking interface to collect a Web resource (number 1), classify it by choosing from a list of pre-defined topics (number 2), receive a set of recommended tags, denoted W_{REC} (number 3), and make a tag assignment (number 4).

recommended tags. It was therefore important to provide a list of topics, which, from the viewpoint of the employees, covered important thematic aspects of the search task and was readily comprehensible for them based on their prior knowledge. To gather such a list, a Web-based idea generation task had been administered one week before study start, asking each employee to "list as many design ideas as possible for a workspace, which could improve the exchange and creation of knowledge in your work group". All listed ideas were then subjected to a qualitative content analysis identifying the most important workspace dimensions mentioned by the employees and reducing these dimensions to the following eight topics: 'Interior design', 'Inspiration sources & techniques', 'Collaboration technologies', 'Gamification & Playfulness', 'Personalization services', 'Augmented reality', 'Wellbeing & health', and 'Socializing'.

Though this list was certainly incomplete, it was exhaustive with respect to the responses the present sample of participants had produced in the idea generation task. To further take account of the opportunity that new topics can be discovered during search, we instructed the participants to inform us whenever they would become aware of an important topic not included in the current list. As we, however, never received such feedback from the participants, the eight-topic list depicted in Figure 4 (number 2) was maintained during the whole study duration.

Measures and statistical analysis

Free association task to observe ideational fluency. To measure ideational fluency, every employee generated free associations for 60 s to each of the eight topics in a Web-based free association task (FAT) at the beginning

(baseline measurement) and after each of the two search conditions (individual and collaborative). Per employee and per point of measurement (baseline, individual, collaborative), we gathered an average FAT response protocol, which was the cumulative number of associations per second averaged across the eight stimuli (topics). In a final step, we merged the average protocols of all 18 employees to characterize the average ideational fluency under each of the three points of measurement in terms of N and λ (by fitting the cumulative exponential of Equation 1).

Tagging consistency. To measure each employee's tagging consistency, we first extracted from the log data all the bookmarks that s/he had collected, where each bookmark was characterized semantically by the set of selected topics (e.g., 'interior design', 'inspiration sources'), denoted T , and by the set of assigned tags (e.g., 'creativity', 'curation', 'mental_fixation') denoted W . Then, for each pair of the employee's bookmarks we calculated two scores, a topic similarity score S_T and a tag similarity score S_W , by applying the *Jaccard* index. Thus, for any two bookmarks x and y , S_T and S_W were given by

$$S_T(x, y) = |T_x \cap T_y| / |T_x \cup T_y| \quad (2)$$

$$S_W(x, y) = |W_x \cap W_y| / |W_x \cup W_y| \quad (3)$$

E.g., if T_x is {'inspiration sources & techniques', 'collaboration technologies'} and T_y is {'collaboration technologies', 'personalization services'}, the intersect and union of both sets include one and three elements, respectively, resulting in $S_T = 1/3$. Above, we defined tagging consistency as the correlation between S_T and S_W . Therefore, to quantify the predicted differences in consistency between the individual and collaborative condition, we performed a regression of S_W on the continuous predictor S_T and the categorical predictor 'Search Condition', and included an interaction term to test the predicted assumption of different slopes under the individual and collaborative condition.

2.3. Results

Our research design and methodology aimed at investigating the assumption that an individual's associative hierarchy becomes flatter during interactions with joint artifacts (i.e., social tags) that are propagated in a shared bookmarking system by virtue of tag clouds. Based on this assumption, we expected a fluency-consistency tradeoff that becomes manifest in a higher ideational fluency (H1.1) and a lower tagging consistency (H1.2) under the collaborative than individual information search.

Hypothesis H1.1

Our first hypothesis H1.1 was that employees exhibit a higher ideational fluency in a free association task (FAT; larger N and smaller λ) after a collaborative than individual information search. Figure 5 presents the average ideational fluency data – cumulative associations (workspace design ideas) generated by 18 employees to eight different topics as a function of

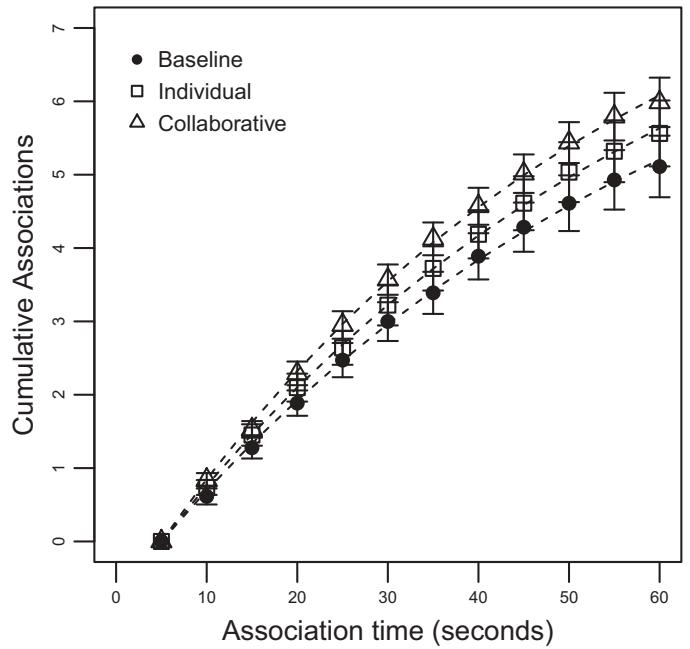


Figure 5. Cumulative free association latency distribution. Error bars represent 1 standard error of mean.

time (seconds) – before the study start (baseline condition; black-filled circles), after the individual search condition (squares), and after the collaborative search condition (triangles). A glance at the three latency distributions reveals a general learning effect of the search task: in comparison to the baseline distribution, employees appeared to produce more associations both after the individual and collaborative search. In addition and according to our expectation, this learning effect seemed to be larger under the latter condition.

In line with research on retrieval from semantic memory (e.g., Rohrer, 2002), the model-based analysis showed that the rate of producing new associations slowed continuously in time and could be well described by the cumulative exponential given by Equation 1. The best-fitting parameter estimates of the exponentials (dashed lines), together with the percent of variance explained, are presented in Table 1. With respect to parameter N , the estimates lend support to Hypothesis H1.1 that this asymptotic number of produced associations increased monotonously from the baseline, over the individual, up to the collaborative condition. Thus, the employees could increase their knowledge especially in the course of the collaborative search, i.e., they could substantially extend the pool of relevant ideas about the eight search topics.

However, in contrast to our expectation, this monotonous increase was not related inversely to the parameter λ , the rate of approach to asymptote, whose estimates were approximately equal for the baseline and individual and, in fact, largest for the collaborative condition. That is, the collaborative search

Table 1. Best-fitting parameter estimates of association latency distributions.

Condition	N	λ	Variance explained
Baseline	8.610	.086	.66
Individual	9.156	.087	.67
Collaborative	9.388	.095	.81

resulted in an increase of both the number and speed of produced associations. In the light of the stochastic search model, this pattern might be explained post-hoc by the assumption that the activation in the mental search set S was not only spread among a larger number of N targets (relevant ideas) but also drained out of the remaining $S-N$ non-targets (irrelevant associations) e.g. through stronger lateral inhibition. In other words, the learning process during collaborative search allowed, on the one hand, getting to know more ideas and on the other hand, effectively inhibiting irrelevant associations or perhaps even replacing them from S . Note that such a pattern still implies a mental activation that is distributed more evenly among a larger number of targets (topic-relevant associations) and that the collaborative search brings about a flatter associative hierarchy for relevant ideas than the individual search.

Finally, a Friedman test on differences in the number of unique associations among the three conditions of baseline (Median = 4.81), individual search (Median = 5.00), and collaborative search (Median = 5.69) reached significance at the .05 level, $\chi^2(2, N = 18) = 5.68, p = .06$. Pairwise comparisons using Wilcoxon test and controlling for the Type I errors by using the LSD procedure further revealed that this effect could be attributed to the difference between the collaborative and baseline condition, $p < .05$; there were no differences neither between the collaborative and individual, $p = .46$, nor between the individual and baseline condition, $p = .45$. In other words, only the collaborative search gave rise to fluency scores that contrasted significantly with those scores that the participants had already been able to achieve before performing the information search.

Summarizing, hypothesis H1.1 assumed ideational fluency to be greater after the collaborative than individual search. As the employees exhibited both the largest number and highest speed of responses under the collaborative condition, we interpret the results as providing support to H1.1. In particular, they harmonize with the assumption of mutual stimulation through joint artifacts, i.e., social tags that act as sign vehicles propagating diverse ideas among the employees. This assumption was further corroborated by a descriptive analysis of participants' click behavior, in particular, their clicks on tags in the shared tag cloud to filter already collected bookmarks within the system. The analysis showed that the probability of a person clicking on a tag that had been introduced by a different person was relatively high, i.e., $p = .68 (SD = .27)$, indicating curiously-driven search behavior and the intentional use of tags to discover novel sources of information. In light of this additional pattern, the active role of tags in mediating the experience of novel ideas appears even more likely. From a learning perspective, this tag-based propagation of ideas is highly desirable, as it seems to broaden and flatten the hierarchy of topic-related associations and thus, supports a more creative encounter with a given topic (e.g., Benedek & Neubauer, 2013).

At the same time, however, a broader and, in addition to that, easily accessible hierarchy of topic-related associations around a given topic can be expected to result in a stronger variability in tag choices for related Web resources. In the following, we therefore investigate this

tradeoff, i.e., the hypothesized downside of a flatter associative hierarchy, namely a higher tagging inconsistency (hypothesis H1.2).

Hypothesis H1.2

Our second hypothesis was that a flatter associative hierarchy under the collaborative condition should become manifest in a more inconsistent tagging behavior, i.e., in a weaker tendency to assign similar tags to topically similar Web resources. In particular, H1.2 was that the relationship between S_W and S_T is stronger under an individual than collaborative search condition. To test H1.2, we first gathered all pairs of an employee's bookmarks, determined each pair's tag and topical similarity (S_W and S_T , respectively; see Equations 2 and 3) and finally, performed a regression of S_W on the continuous predictor S_T and the categorical predictor 'Search Condition'.

685 data points entered the regression, which consisted of the 361 bookmark pairs of the individual and of the 324 bookmark pairs of the collaborative condition. The model explained about 13% of variance in the tag similarity S_W of an employee's pair of bookmarks (adjusted $R^2 = 0.132, p < .001$) and yielded a highly significant effect for the predictor S_T ($t = 9.87, p < .001$), and – in line with H1.2 – a highly significant interaction between this continuous predictor and the categorical predictor 'Search Condition' ($t = -4.44, p < .001$). Table 2 shows the estimates of the model's intercept and slope and how these estimates change as a function of 'Search Condition'. The small amount of variance explained is not surprising as the probability of reusing tags (that underlies S_T) depends not only on semantic attributes of the resources, but also on mediating cognitive processes (e.g., Fu & Dong, 2012; Seitlinger et al., 2015), which are to some extend specified by the tag recommender's algorithm presented in the article's second part. The present regression model, however, did not capture such cognitive processes because its primary purpose was not to explain a large amount of variance in the individuals' tagging behavior but to determine whether the amount of variance explained by the predictor S_W differs between the individual and collaborative condition.

Under the individual information search, the standardized coefficient β_1 (i.e., the slope of the predictor S_T) takes on the value of about 0.40, which is indicative of a moderate effect size. Thus, the higher S_T , i.e., the higher the topical similarity of two Web resources collected by an employee, the higher is the similarity of the tags S_W applied to that pair of Web resources. However, as Table 2 also shows, the predictor's slope significantly declines under the collaborative condition and, in fact, takes on the value of only 0.14, which is indicative of a small effect size. Thus, the small amount of variance explained (ca. 13%) in the whole dataset can probably be attributed to the fact that S_T is a substantial predictor of S_W only under the condition of an individual information search. These results are well in

Table 2. Summary of the regression of S_W (tag similarity of a bookmark pair) on S_T (the pair's topical similarity) and 'Search Condition'.

Search condition	Intercept	Slope
Individual	$\beta_0 = 0.05$	$\beta_1 = 0.39$
Collaborative	$\beta_0 + \alpha = 0.12$	$\beta_1 + \beta_2 = 0.14$

Note. $\alpha = 0.07; \beta_2 = -0.25$

accordance with H1.2, which assumes that under the collaborative condition, where tag clouds propagate divergent tag-topic associations, the tendency to reapply the same tags for reoccurring topics (semantically similar Web resources) is smaller than under the individual condition.

Controlling for priming effects.¹ Especially with respect to hypothesis H1.2 (on differences in participants' tagging consistency), a question may be to what extent our result patterns were an artifact of having participants select from the eight pre-defined topics, as this potentially had priming effects on their subsequent choice of tags. However, as our main interest was in investigating the impact of an experimental variable (differing search conditions), we did not expect our results to be sensitive to the topic structure because topic-based priming should not vary between the conditions. Nevertheless, in order to explore its potential impact, we performed a statistical control analysis, which is described next.

We performed a second regression analysis by entering the categorical predictor 'topic use'. Based on a median split, 'topic use' distinguished between participants who on average selected many vs. few topics to describe a given bookmark. The rational behind it was that if topic-based priming effects were actually negligible for our results (on condition-dependent differences in consistency), hypothesis H1.2 (stronger $S_T - S_W$ relationship under the individual than collaborative condition) should hold in both pre-experimental groups that can be assumed to be affected by different priming effects due to different ways of interacting with the topic structure.

The dummy variable split the sample along the median of 1.8 into a "few topics" and a "many topics" group that on average had selected 1.44 and 2.46 topics per resource. This extended regression model indeed yielded a second-order interaction ($t = 2.51, p < .05$), which, however, was of an ordinal nature and only qualified our previous data interpretation: under both groups, the relationship between S_T and S_W was stronger under the individual than collaborative search, where this difference was greater in the "few topics" (standardized coefficients: $\beta_{\text{individual}} = .42, \beta_{\text{collaborative}} = .05$) than "many topics" group ($\beta_{\text{individual}} = .31, \beta_{\text{collaborative}} = .23$). This effect can be explained by drawing on the search of memory model (Figure 2): selecting more topics provides more context cues, which constrain the mental search set more strongly and in further consequence, reduce the variance in tagging behavior across time and different environmental conditions, such as those of an individual vs. collaborative search. Summarizing, the regression model extended by the variable of 'topic use' further strengthened hypothesis H1.2 because the experimentally induced difference in consistency could be observed in both pre-experimental groups. We therefore conclude that topic-based priming processes did not contribute substantially to the observed differences in search condition. Hence, our results should generalize to other settings that e.g. make use of a more or less differentiated topic structure.

2.4. Discussion

We gathered evidence for the expected fluency-consistency tradeoff by observing opposing effects of 'Search Condition' on temporal dynamics of free associations and tagging behavior: Displaying social cues (under the collaborative condition) increased ideational fluency (measured by a FAT), but reduced the consistency of tag choices for reoccurring topics (measured by a correlational analysis of log data). Networked search (e.g., Lazer & Bernstein, 2012), however, benefits from balancing individual exploration – a process that benefits from ideational fluency (e.g., Kerne et al., 2014) – and exploitation of each other's search results, which, in the context of tag-based bookmark sharing, is facilitated by topically consistent tag choices (Dellschaft & Staab, 2012). Therefore, finding evidence of the fluency-consistency tradeoff begs the question of how we are to deal with the downside of a flat associative hierarchy, i.e., growing inconsistency.

One potential answer will be explored in the next section where we have contrasted the use of two tag recommendation mechanisms (TRM). One of those mechanisms is built on a "Most Popular Tag (MPT)" recommendation strategy, a variant of which can be found in many contemporary web environments. The other is derived from a mechanism that models associative hierarchies over different search topics and produces tag choices as a search of memory. As the latter is especially tuned to a situation of flat associative hierarchies and increased ideational fluency, it should be especially effective in the collaborative information search condition.

3. An effective recommendation mechanism for collaborative information search

Tag recommendation mechanisms (TRM) can be seen as non-human actors (Webster, Gibbins, Halford, & Hracs, 2016) to which developers of social information systems delegate the intention of making information search more effective. TRM can be directed either toward convergence (i.e., tagging consistency) for the sake of aligning the vocabulary (e.g., Font et al., 2015), or divergence (i.e., ideational fluency) for stimulating creative ideation (e.g., Kerne et al., 2014). For system designers, this means it is difficult to achieve both things at the same time.

When designing TRM, we draw on our framework of reflective search. By modeling a fundamental search process, namely that of search of (human) memory when reflecting on a resource, it allows to deal with cognitive dynamics involved in the tradeoff between exploration and exploitation of networked search. We introduce SoMe, a TRM that simulates search of memory dynamics in response to a resource and hence, to identify tags that are co-created in collaborative search and are likely to resonate with a user's current reflection. That way, the probability of reusing tags for re-occurring topics (i.e., tagging consistency) can be increased and the tradeoff balanced toward exploitation when associative hierarchies are flat (as in the situation of collaborative search): in this situation, inconsistent tag choices for reoccurring topics

¹We thank an anonymous reviewer for prompting this additional analysis.

may impair convergent group processes, i.e., reduce the benefits of exploiting each other's search results through a consistent tag vocabulary. We then compare SoMe to a most popular tags (MPT) recommender. SoMe's advantage over MPT should come to the fore under the collaborative search condition, that is, if the relationship between word and meaning (tag and topic) becomes looser (less consistent) due to flatter associative hierarchies.

In what follows, we first demonstrate this decreased semantic distinctiveness (looser tag topic relationship) in collaborative information search. This decreased semantic distinctiveness of popular tags should lead to problems when trying to predict and recommend tag choices by estimates of popularity, as in the case of MPT approaches.

In a second step, we then describe in detail how SoMe simulates a person's search of memory when choosing tags in order to filter an inconsistent tag vocabulary by searching for topically resonant (relevant) tags. Finally, we provide the results of the evaluation that has taken place in the same study reported above and addresses the second research question *whether SoMe achieves high tag acceptance rates under a collaborative search and, more specifically, whether the SoMe advantage (over MPT) is larger under the collaborative than individual search condition.*

3.1. The challenge of predicting tags based on flat associative hierarchies

Above, we have shown the shape of an associative hierarchy to affect tagging consistency, where a steep hierarchy gives rise to a closer relationship between tag and topic than a flat one (see Table 2). Furthermore, we argue that the extent of tagging consistency is related to the extent of semantic distinctiveness of popular tags (to be shown below), which in turn should determine the extent by which a TRM has to combine statistics of both popularity (tag use frequency) and semantic resonance (topical relevance of a tag) to derive appropriate recommendations.

For instance, in case of a highly consistent tagging behavior, a given tag w_i can be expected to be chosen quite exclusively for a certain topic t_j . More formally, if $P(t_j|w_i)$ denotes the posterior probability that w_i belongs to t_j , it should exhibit high estimates for only one of the eight search topics (of the current study) and rather low estimates for the remaining seven topics, given w_i is used consistently. The posterior probability can be calculated applying the Bayes theorem (e.g., Fu & Dong, 2012), where, in this study, the priors are estimated by counting their relative frequencies being assigned to bookmarks, and the conditional probability $P(w_i|t_j)$ by dividing n_j (number of bookmarks in t_j that are associated with w_i) by n (total number of bookmarks in t_j). To observe the shape of the average distribution of the posteriors over the eight topics (e.g., 'peaky' in case of consistency and flat in case of inconsistency), per tag we can – similar to a rank-frequency distribution – rank the eight topics according to their strengths, i.e., $P(w_i|t_j)$, then average these ranked posteriors across several tags (e.g., the seven most popular tags, MPT), and finally, as shown in the left diagram in Figure 6, draw these means against the corresponding ranks. In line with our expectation, the left diagram reveals that $P(w_i|t_j)$ drops off comparatively steeply under the individual search (solid line), which means that the steep associative hierarchies (reported above) are reflected in the emerging tag vocabulary by a steep hierarchy of the posteriors: given a particular topic, we can predict the choice of tags with higher certainty. By contrast, under the collaborative search, the flat associative hierarchies become manifest in a comparatively flat distribution of the posteriors (dashed line), indicating higher uncertainty when predicting tags based on a given set of resource topics.

As already argued, the strength of relationship between tag and topic should further affect the extent to which popular tags are topically distinct from each other. To quantify distinctiveness D_i of a tag w_i , we introduce the notion of a tag's semantic profile S_i , which is simply the unranked distribution of $P(w_i|t_j)$ over the eight topics. D_i can then be defined as 1 minus the average cosine similarity between S_i and each of the

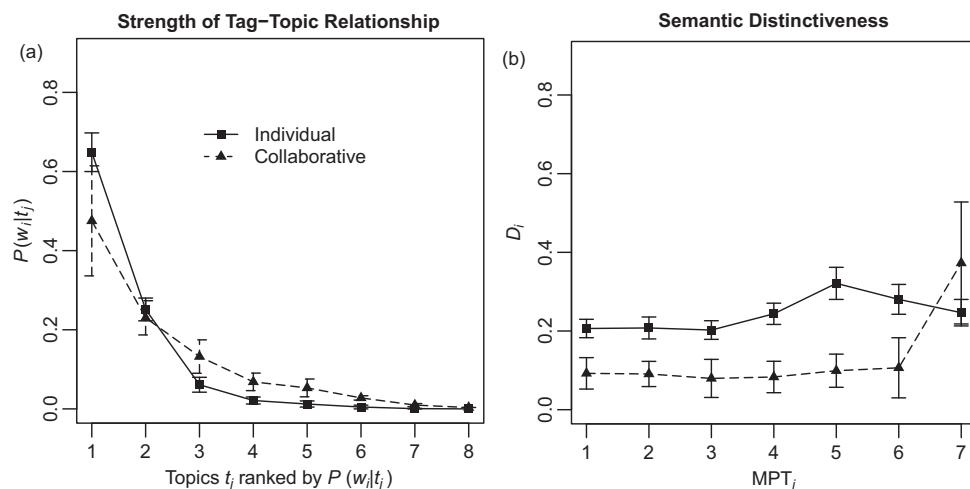


Figure 6. Strength of tag-topic relationship (left diagram, Figure 6a) and semantic distinctiveness (right diagram, Figure 6b) as a function of "Search Condition" (Individual vs. Collaborative). Error bars represent 1 standard error of mean.

profiles of the remaining six tags in MPT. In accordance with our expectation, the right diagram in Figure 6 shows D_i to be pronounced more strongly under the individual than collaborative search (except for the seventh tag in MPT), indicating a less ambiguous tag-topic co-occurrence pattern under the individual search, where different popular tags seem to refer to different search topics.

Summarizing the two patterns of Figure 6, we anticipate that the strategy of an MPT-based TRM, i.e., recommending tags from the head of the rank-frequency distribution, should be effective under the individual search condition: Due to its semantic distinctiveness and high popularity, the set of MPT (head of the rank-frequency distribution) is likely including a subset of familiar tags that fit the unknown resource an employee is bookmarking and tagging. Under the collaborative condition, however, statistics of popularity (rank and frequency) do not appear to provide sufficient information to disambiguate the tag recommendation problem: sampling from MPT does not necessarily result in a set of semantically distinct tags; instead it is likely to obtain a set of prominent tags indexing the same or similar topics. Thus, to identify tags that are both popular and topically distinct, a TRM strategy is needed filtering the tag vocabulary by popularity and semantic resonance (relevance).

3.2. Modeling search of memory to disambiguate the tag recommendation problem

To realize a TRM that searches for popular and semantically resonant (topically relevant) tags, we make use of a strategy that has been conceptually proposed (but not empirically evaluated) by Seitlinger, Ley, and Albert (2013). First simulation-based analyses of large-scale social tagging datasets (Kowald et al., 2014) have shown this strategy to be successful in modeling and predicting users' tag choices. The question, however, whether these results (i.e., high prediction accuracies) generalize to a realistic information search scenario, and whether tags can be recommended the employees actually adopt for their tag assignments, remains open and is addressed in the following.

This strategy implements the search of memory (SoMe) scheme (Figure 2), i.e., the cue-based activation of a search set S (step 1) and subsequent selection of N cue-relevant associations (tags; step 2). Drawing on MINERVA2, a model of episodic memory (e.g., Hintzman, 1986; Kwantes, 2005; Sprenger et al., 2011), the SoMe recommender distinguishes a primary memory (PM) that represents the experienced retrieval cue (e.g., the resource being bookmarked and tagged) from a secondary memory (SM) – “the vast pool of largely dormant memory traces” (Hintzman, 1986, p. 412). In the present study, the retrieval cue in PM is represented as the subset of search topics the employee assigns to the present resource via the bookmarking interface (Figure 4); a memory trace in SM is a record of each of the employee's bookmarks, in particular, of the correspondingly chosen topics and tags. The set of all SM traces therefore provides a memory-like representation of employee-specific topic-tag associations.

In the first step of the simulated PM-SM communication (cue-based activation of S), all traces in SM (topic-tag

associations) are activated in parallel – according to their topical similarity to the cue (i.e., topic subset in PM): the more topics the cue and the trace have in common, the higher the trace's activation. That way, we implement a simple mechanism of semantic resonance, by which those tags come to the fore that are associated with cue-relevant topics. In the second step, the SoMe recommender performs a frequency-based ranking within the subset of strongly activated traces in order to select N cue-relevant associations (tags).

Summarizing, SoMe can be regarded a semantically resonant MPT approach that should improve MPT substantially, if semantic distinctiveness among MPT (see Figure 6) is low due to flat associative hierarchies. Based on the results of the first part that associative hierarchies are flatter and ideational fluency is higher under the collaborative than individual search, the third hypothesis H2 is: *The proposed tag recommender SoMe reaches higher levels of acceptance rate, i.e., is more likely recommending tags the user actually adopts for personal tag assignments, than MPT during a collaborative information search; no differences should be observed under the individual search condition.*

3.3. Method

This hypothesis (H2) was explored in the same study as previously reported.

Design

To answer the second research question and test hypothesis H2, the design (already described above) also included the independent variable ‘Type of TRM’ (MPT vs. SoMe; within subjects): When an employee bookmarked and tagged a new Web resource, (s)he could choose from a set of recommended tags that was generated either by MPT or SoMe – with an equal chance for either TRM being applied. Thus, to find an answer to our second research question, we realized a 2 (Search Condition: individual vs. collaborative search) \times 2 (Type of TRM: MPT vs. SoMe) research design. The second dependent variable was acceptance rate of the tags generated by the two TRMs. Each of these variables and its operationalization are described below (see Section Measures). For the sake of a comprehensible presentation, however, we first describe some more details on the search task as well as the technical infrastructure.

Participants and search tasks

The process of annotating a bookmark, i.e., choosing topics and assigning tags, was supported by the bookmarking interface, illustrated by Figure 7, which this time also schematizes the tag recommenders' underlying algorithm (represented by the flow chart on the right-hand side): After selecting from the eight-topic list (number 2) to specify T (i.e., the subset of chosen topics) and clicking on the ok button, the algorithm was initiated producing a set of recommended tags, denoted W_{REC} : Depending on the outcome of the algorithm's first step, namely the random decision to choose between one of the two recommenders, W_{REC} was based on either the tags' frequency counts alone (in case of MPT) or on both the tags'

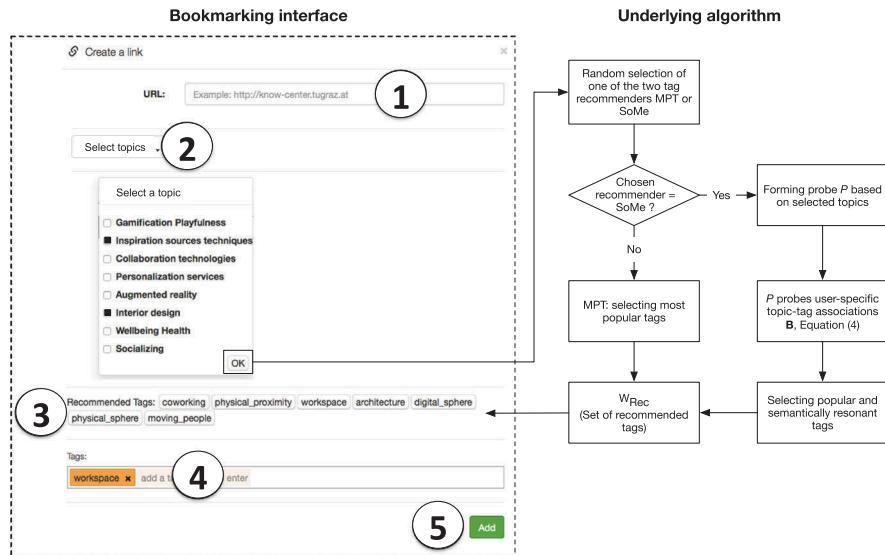


Figure 7. Bookmarking interface (left-hand side) and schematic illustration of the algorithm generating the set of recommended tags W_{REC} (right-hand side).

frequency counts and their semantic relatedness to T (in case of SoMe). The semantic relatedness was derived from a participant's past search behavior, in particular from topic-tag choices assigned to previously collected bookmarks. Details on how these choices were represented and processed by SoMe are given in the next section.

Finally, after displaying W_{REC} (number 3), the participant was free to choose from the recommended tag list as well as to type in personal tags (number 4). Note that this sequence of steps – encompassing the selection of topics, the algorithm, and the choice of tags – ensured that there were no interaction differences (from a user perspective) between the two recommenders, even if only SoMe was actually exploiting the topical information given by T .

Recommendation mechanisms

SoMe. The Search of Memory (SoMe) tag recommendation mechanism (TRM) was designed to mimic basic principles of an employee's search of memory when tagging a current Web resource. To this end, SoMe is based on the episodic memory model MINERVA2, which has been shown to account for a wide range of memory-based human behavior, such as recognition (Hintzman, 1988), categorization (Hintzman, 1986), representing word similarities (Kwantes, 2005) or judgments on the probabilities of hypotheses (Sprenger et al., 2011). Our goal, of course, was not to represent a user's entire episodic memory but to account for the encoding of episodic memory traces in the course of the user's ongoing search history. Beyond that, we did not aim to model specific search operations that would be formalized by more detailed models, such as CMR (e.g., Polyn, Norman, & Kahana, 2009), but to mimic the general principle of activating memory traces that are similar to a given environmental cue, i.e., the resource being tagged. To model this general principle, SoMe implemented the MINERVA2 distinction between a primary memory (PM) and secondary memory (SM) component. The role of PM was to represent the resource being tagged in terms of the chosen topics and to search SM for tags that the employee had assigned to

bookmarks with similar topics. To model PM, a probe P was formed as an 8-element, binary (1,0) feature vector – with each feature j representing one of the eight topics and taking on the value +1, if it was assigned to the current resource. SM was represented as a matrix B , with each row i being a binary (1,0) feature vector representing a bookmark in the employee's previous information search. The first $j = 1, \dots, 8$ positions indexed the topic-features, the subsequent $j = (8+1), \dots, (8+n)$ positions indexed the tag-features, where n was the number of tags generated within the whole bookmarking system.

To mimic a search of memory, SoMe proceeded in two steps: First, in a process of resonance, P interacted with the topic features ($j = 1, \dots, 8$) of each row i in B to generate an overall value of activation $A(i)$ given by the cubed cosine between P and the row's first 8 feature values. Second, the mechanism estimated $R(j)$, i.e., the extent to which a feature resonated with the probe, by multiplying all features of each trace by the trace's activation $A(i)$ and then, summing these products across all m traces, given by

$$R(j) = \sum_{i=1}^m A(i)B(i,j) \quad (4)$$

Under the individual search condition, the tags, i.e., $j = (8+1), \dots, (8+n)$, with the 7 highest resonance values $R(j)$ were included in W_{REC} and displayed in the bookmarking interface (number 3 in Figure 7). Under the collaborative condition, an average resonance value $\bar{R}(j)$ was calculated by mimicking a search of each employee's memory B and averaging across the individual $R(j)$ values. Finally, the seven highest ranked $\bar{R}(j)$ values were included in W_{REC} .

MPT. This type of TRM only considered the reuse count of tags and thus, determined the current rank-frequency distribution of all tags that had been generated in the shared (collaborative) or unshared (individual) bookmarking system. Then, the seven highest ranked tags were included into W_{REC} and displayed in the bookmarking interface.

Measures

Tag acceptance rate. The tag acceptance rate was measured calculating the precision and recall of the recommendations generated by MPT and SoMe. Each time an employee collected a new bookmark and classified it by selecting a set of topics, the bookmarking interface (Figure 7) displayed a set of $x = 7$ recommended tags (number 3 in Figure 7), denoted W_{REC} , and the employee was free to type in personal tags and/or to choose tags from W_{REC} . If W_{APP} denotes the set of tags actually applied by the employee, the metrics of *precision* and *recall* are calculated by the following two fractions:

$$\text{precision} = |W_{APP} \cap W_{REC}| / |W_{REC}| \quad (5)$$

$$\text{recall} = |W_{APP} \cap W_{REC}| / |W_{APP}| \quad (6)$$

To combine both metrics and to derive a single score, the harmonic mean of *precision* and *recall* is calculated, which is denoted *F-score* and given by

$$F = (\text{precision} * \text{recall} * 2) / (\text{precision} + \text{recall}) \quad (7)$$

Note that these metrics depend on the number of elements (tags) included in W_{REC} , and thus, precision, recall and the *F-score* are usually determined for each possible x , which varied between $x = 1$ and $x = 7$ in the present study. Following this evaluation practice, we determined the average *F-score* under the four factorial combinations (of the 2×2 research design) and for every x . For statistical analysis, we then performed a 2 (Search Condition) \times 2 (Type of TRM) repeated measures ANOVA on the *F-score*, taking x as the unit of analysis.

3.4. Results

We expected an interaction of the search condition with the performance of the tag recommender SoMe, i.e., a larger advantage over the baseline recommender MPT under the collaborative than individual condition (H2).

Hypothesis H2

The tag recommender SoMe was designed to improve the recommendation of tags under a collaborative condition, i.e., conditions of flat associative hierarchies and low semantic distinctiveness. We, therefore, hypothesized an interaction between ‘Type of TRM’ (MPT vs. SoMe) and ‘Search Condition’ with respect to the tag acceptance rate and in particular, a larger SoMe advantage under the collaborative condition. To test this interaction, we extracted all tagging events under each of the four factorial combinations and compared the set of recommended tags (W_{REC}) with the set of actually applied tags (W_{APP}) by calculating the *F-score* (Equation 7). Figure 8 presents the results. Under the individual condition (left diagram), the two recommenders appear to reach similar estimates of the *F-score* for a varying number x of recommended tags (drawn on the abscissa). Averaged over x , the *F-scores* achieved by MPT and SoMe are 0.29 ($SD = 0.04$) and 0.30 ($SD = 0.03$), respectively. We therefore conclude that during an individual information search, where employees exhibit comparatively steep associative hierarchies, both MPT and SoMe generated recommendations at a comparable acceptance rate.

As expected, the relation between the two recommenders changed under the collaborative condition (right plot), where, descriptively, SoMe reached higher estimates of the *F-score* than MPT across all values of x . In this case, the average and x -independent scores for SoMe and MPT are 0.34 ($SD = 0.09$) and 0.27 ($SD = 0.08$), respectively. Thus, we observed an interaction between the variables ‘Search Condition’ and ‘Type of TRM’ and found that SoMe outperformed MPT only during a collaborative information search. The results of the ANOVA supported this interpretation by yielding a significant effect for the interaction of the two variables, $F(1,6) = 12.45$, $p < .05$. Beyond that, the test yielded a significant main effect for ‘Type of TRM’, $F(1,6) = 53.96$, $p < .001$, but not for ‘Search Condition’, $F(1,6) = 0.19$, *n.s.* Due to the interaction, we did not consider the main effect and concluded in line with hypothesis H2 that the SoMe-advantage over MPT only applies to the collaborative condition.

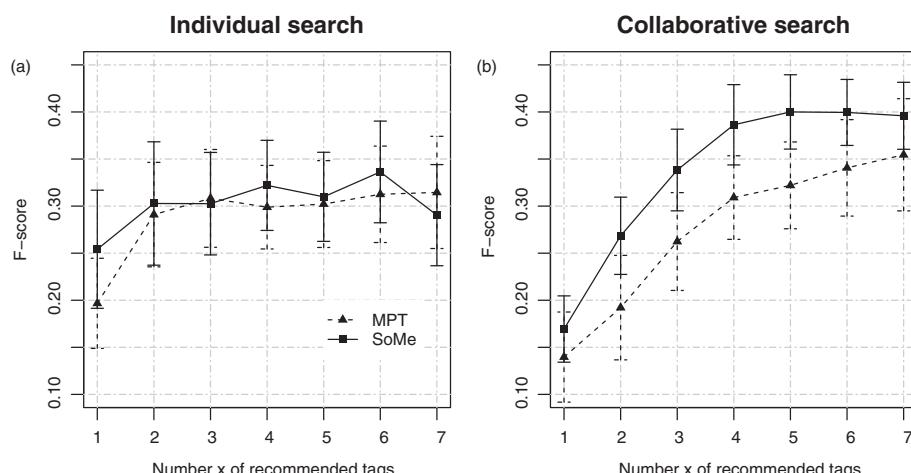


Figure 8. Tag acceptance rate achieved by the MPT and SoMe for a varying number of recommended tags ($x = 1, \dots, 7$) under the individual (left, Figure 8a) and collaborative search (right, Figure 8b). Error bars represent 1 standard error of mean.

3.5. Discussion

These results demonstrate that the MPT-based approach is less appropriate to predict and support individual tagging behavior, if it is embedded into a collaborative search scenario where associative hierarchies tend to be flat. A considerably more effective strategy to derive topically relevant recommendations under inconsistent tagging patterns appears to be the simulation of a resource-triggered search through associative memory for familiar and semantically resonant words.

Therefore, the proposed recommender SoMe seems to implement a promising and robust approach to increase employees' tendencies of reusing existing tags independent of how flat or steep the underlying associative hierarchies are. Hence, it appears to be an effective and expandable strategy for a non-human actor (i.e., TRM) helping to balance exploration/divergence and exploitation/convergence during a networked search. This is particularly the case in a collaborative situation, where ideational fluency is high (driving individual exploration and experimentation) and tagging consistency needs to be supported to also drive exploitation of each other's search results. In such situation of higher inconsistency, a given topic is more likely to co-occur with several tags with similar frequencies. As a consequence, the rank-frequency distribution of the tag vocabulary becomes flatter and popular tags (head of the rank-frequency distribution) start overlapping in the topics they are used for. Thus, statistics of popularity (e.g., usage frequency) alone no longer suffice to identify topically distinct popular tags, among which a subset would be likely to match a current resource's topics. In contrast, SoMe overcomes this problem of decreased distinctiveness by filtering the tag vocabulary not only by popularity but also semantic resonance.

By doing so, SoMe considers more information about an employee's tagging behavior than MPT does, and therefore a general advantage of SoMe over MPT (a "main effect") is of course expected. However, the question we have explored with our study is not whether SoMe in general is a more effective recommender than MPT, but rather whether the advantage of SoMe interacts with the Search Condition, i.e., whether the advantage is larger in the collaborative search condition. We are also not suggesting SoMe as the most effective recommendation strategy in collaborative information search. To be able to claim this would require a broader set of alternative strategies to compare SoMe to. In this article, we were rather interested in examining some of the effects of collaborative information search on more fundamental cognitive processes such as memory retrieval.

4. Overall discussion

The first goal of the study was to investigate the question whether mutual stimulation during a collaborative and Web-based information search has an impact on an employee's mental organization of associations around a given topic. In particular, we expected a tag-mediated circulation of ideas (collaborative search) to let an employee's associative hierarchy become flatter, which should be indicated by a tradeoff between an increase in ideational fluency and a decrease in

tagging consistency. The second goal was to introduce and test the tag recommendation mechanism SoMe designed to push the reuse of tags and thus, to compensate for the hypothesized inconsistency in a collaborative search that benefits from balancing processes of divergence (e.g., fluency) and convergence (e.g., consistency; Lazer & Bernstein, 2012).

First, the task of searching for Web resources was found to yield a general learning effect independent of the Search Condition (individual vs. collaborative) by extending the associative structure by topic-relevant representations. Additionally, as this increase of associations did not reduce their availability (response time), we concluded that learning consisted not only in broadening the fan of topic-relevant associations (i.e., flattening the associative hierarchy) but also in inhibiting or even excluding topic-irrelevant associations. The resulting associative structure gave rise to an increased ideational fluency, which was characterized by both a larger number and a higher speed of responses. Beyond that, this effect appeared to be larger under the collaborative than individual condition and became manifest in a fluency-consistency tradeoff. In line with hypothesis H1.1., the flatter and more topic-related hierarchy under the collaborative condition allowed for a steadier stream of ideas in response to a particular topic (e.g., 'interior design'); at the same time, it also caused a more variable assignment of tags to re-occurring topics during Web search, thereby lending support to H1.2.

From a psycho-pedagogical perspective, we therefore gained evidence of a positive impact of mutual stimulation on individual learning processes: a higher ideational fluency facilitates creative cognition to the extent that it increases the probability of bringing otherwise dissociated ideas for a new and useful combination into ideational contiguity (e.g., Benedek & Neubauer, 2013; Mednick, 1962). Also from an information discovery (Kerne et al., 2008) and information-based ideation viewpoint (Kerne et al., 2014), our results have practical design implications by revealing that an individual is more likely to experience cognitive restructuring and to forge new associations when searching the Web, if she or he participates in a collaborative and tag-based curation of an evolving problem space. From the perspective of information retrieval, however, a tradeoff on the expense of consistency exacerbates the stabilization of the tagging vocabulary and thus, organization of an already curated space.

In order to compensate for this downside of mutual stimulation and to support a tag-based collaborative information search, we deduced and evaluated the tag recommendation mechanism (TRM) SoMe from the search of memory scheme. By mimicking a resource-triggered search of associative memory (topic-tag associations) to identify and recommend topically relevant tags, SoMe achieved adequate tag acceptance rates and, under the collaborative condition, outperformed a baseline TRM built upon the most popular tags (MPT) approach (hypothesis H2). The SoMe advantage during a collaborative search provided evidence of the model's validity (in representing an associative hierarchy) and showed an increased availability of a larger number of associations to be mirrored by a looser tag-topic relationship. Under such conditions of inconsistency, a tag's topical relevance (modeled

by the search through memory) had to be taken into account in addition to the tag's popularity (modeled by frequency) in order to anticipate an employee's tagging behavior accurately. Again our model of reflective search would predict such interaction: Under the individual search condition, TRM and tag clouds create a self-reinforcing loop between the individual and her or his own artifacts that strengthens tag associations to only a few and recurrently reflected search topics. The consequence is a steep associative hierarchy with semantically distinct tags ranking at the hierarchy's top. Under the collaborative condition, however, TRM and tag clouds take a propagating role and animate an individual to reflect and experience tags in a drifting context of varying topics. The consequences are broader and overlapping fans of topic associations that are hardly distinguishable based on popularity.

5. Conclusions and practical implications

The study reported here adds further evidence to the reflective search framework that we have developed in the context of social tagging on the web. Through this study, we have added evidence through two diverse research methods. First and in a more traditional way, we test the effects of information search on mental organization (associative hierarchy). By modeling and simulating memory access mechanisms with a TRM, we then gather further evidence for the soundness of the approach by means of an entirely new research strategy. Taken together, we believe these results offer convergent evidence for the framework we have been proposing.

This approach is similar to that of Nijstad and Stroebe (2006) in their research on group creativity. Like these authors, we show how a model of learning and search of memory can become part of a socio-cognitive framework (i.e., reflective search) and how this integration helps to describe and simulate supra-individual processes of tag-based mutual stimulation. In our understanding, social tags that people encounter in a shared environment do not lead to imitation, but rather trigger a reflection of the search topic which changes their mental structures. The model of search through memory (see Figure 2) serves as starting point to specify the notion of an associative hierarchy that we assume to be enacted when users reflect on objects (e.g., problems, Web resources; Seitlinger & Ley, 2016). The evidence we found for changes in that associative hierarchy lends further evidence to the assumption that reflection triggers an individual learning process (e.g., Renner, Prilla, Cress, & Kimmerle, 2016). This act of "learning-by-reflecting" is stimulated by joint artifacts (e.g., social tags) and thus, contributing to a collaborative search results in a broader fan of topic-related associations and consequently, an increased level of ideational fluency.

As ideational fluency is an indicator of creative potential (e.g., Benedek & Neubauer, 2013), the present study also sheds some light on a process that appears to be conducive to group creativity (e.g., Nijstad & Stroebe, 2006; Paulus & Brown, 2007), namely artifact-mediated learning-by-

reflecting, and how this process can be facilitated by means of creativity-support tools (e.g., Kerne et al., 2008; Sarmiento & Stahl, 2008), namely tagging combined with an effective tag recommendation mechanism (TRM). Especially in the light of echo chamber effects (Pariser, 2011; Sunstein, 2001), which are frequently observed in online discourses, intelligent information services become more important that prevent a group from converging in a particular perspective too early. In future work, we therefore would like to investigate the question whether the applied recommender approach of providing stimuli that are novel but also resonate with personal knowledge structures can be an effective way to maintain a distribution of diverse ideas and thus, to counteract echo chamber effects.

The present work also has implications for the application of memory models in the context of TRM. While in the present work, an abstract scheme of memory search and a simple computational interpretation sufficed to account for a comparatively simple behavior (i.e., choosing tags), continuative research questions on Human-Web interactions could require more complex models of memory search. The present study took into account two landmarks of memory psychology in recommending tags, namely *familiarity* (represented by the popularity of tags), and *semantic resonance* (as specified by the topical filtering through the MINERVA model). In the future, one goal will be to further refine the SoMe strategy, for instance, by extending the model by *recency* of tag use (e.g., Trattner, Kowald, Seitlinger, Kopeinik, & Ley, 2016), another important factor in retrieval from memory. Furthermore, context retrieval models (e.g., Polyn et al., 2009) would allow specifying and parameterizing executive processes (e.g., internal context updates) that take part in higher-level cognition (e.g., reflection and formation of information goals) during Web exploration (Seitlinger & Ley, 2016).

6. Limitations and future work

One limitation of this study is the small sample size of $N = 18$ participants, which is owed to the difficulty of acquiring employees motivated to take part in a four-weeks work-integrated field study. To face this problem, we took particular statistical measures inspired by previous studies on social tagging, such as the one of Pirolli and Kairam (2012) or the one of Fu and Dong (2012) who performed model-based analyses of data from samples of $N = 18$ and $N = 8$ participants, respectively. For instance, to investigate hypothesis H1.1, we drew on a well-established modeling technique of memory psychology, which allowed deriving robust estimates (of N and λ) by aggregating responses across both different stimuli (i.e., topics) and different participants. Furthermore, to examine H1.2, not the individual participant but her or his recorded activities within the bookmarking system served as units of analysis, resulting in 685 rather than 18 data points. Finally, for analyzing H2, the unit of analysis was x (i.e., the number of recommended tags), which again allowed for aggregating across participants and all their

bookmarks and hence, deriving robust estimates of their tag acceptance rates.

While these statistical techniques help mitigate the problem of small sample sizes, they do not dissolve it. Therefore, future studies are necessary that clarify whether the reported results can be replicated and whether they generalize to different conditions. On the one hand, such studies will have to explore the impact of participants' background and level of expertise in searching for information. The present sample consisted of knowledge workers who can be assumed to be highly trained in performing the activities addressed by our research questions. Thus, even if we assume general-psychological processes are at play, such as cue-dependent search of memory, which should generalize across knowledge domains and skill levels, questions around external validity remain and must be clarified empirically. On the other hand, generalizability will have to be checked in terms of software features that create affordance for particular behavior. For instance, the eight-topic list of the bookmarking interface could have prompted specific priming processes when participants assigned tags to their collected resources and thus, affected their tagging consistency. Though statistical control analyses indicated no substantial impact arising from the number of chosen topics, future experiments are necessary to systematically observe the effect of a more or less differentiated upper-level structure.

Disclosure of potential conflicts of interest

No potential conflict of interest was reported by the author.

Funding

This work was supported by the Austrian Science Fund (FWF); [P25593-G22, P27709-G22]; and by the European Union projects Learning Layers; [318209]; CEITER; [669074].

References

- Albert, D. (1968). Free recall of word sequences as stochastic storage removal. *Zeitschrift Für Experimentelle Und Angewandte Psychologie*, 15, 564–581.
- Benedek, M., & Neubauer, A. (2013). Revisiting Mednick's model on creativity-related differences in associative hierarchies. Evidence for a common path to uncommon thought. *The Journal of Creative Behavior*, 47(4), 273–289. doi:[10.1002/jocb.35](https://doi.org/10.1002/jocb.35)
- Bousfield, W. A., & Sedgwick, C. H. W. (1944). An analysis of restricted associative responses. *Journal of General Psychology*, 30(2), 149–165. doi:[10.1080/00221309.1944.10544467](https://doi.org/10.1080/00221309.1944.10544467)
- Bousfield, W. A., Sedgwick, C. H. W., & Cohen, B. W. (1954). Certain temporal characteristics of the recall of verbal associates. *American Journal of Psychology*, 67(1), 111–118. doi:[10.2307/1418075](https://doi.org/10.2307/1418075)
- Candy, L., & Hewett, T. (2008). Special issue introduction: Investigating and cultivating creativity. *International Journal of Human-Computer Interaction*, 24(5), 441–443. doi:[10.1080/10447310802142060](https://doi.org/10.1080/10447310802142060)
- Davelaar, E. J., & Raaijmakers, J. G. W. (2012). Human memory search. In P. M. Todd, T. T. Hills, & T. W. Robbins (Eds.), *Cognitive search* (pp. 177–194). Cambridge, MA: MIT Press.
- Dellschafft, K., & Staab, S. (2012). Measuring the influence of tag recommender on he indexing quality in tagging systems. In E. Munson & M. Strohmaier (Eds.), *Proceedings of the 23rd ACM conference on hypertext and social media* (pp. 73–82). New York, NY: ACM Press. doi:[10.1145/2309996.2310009](https://doi.org/10.1145/2309996.2310009)
- Dorfman, L., Martindale, C., Gassimova, V., & Vartanian, O. (2008). Creativity and speed of information processing: A double dissociation involving elementary versus inhibitory cognitive tasks. *Personality and Individual Differences*, 44(6), 1382–1390. doi:[10.1016/j.paid.2007.12.006](https://doi.org/10.1016/j.paid.2007.12.006)
- Font, F., Serrà, J., & Serra, X. (2015). Analysis of the impact of a tag recommendation system in a real-world folksonomy. *ACM Transactions on Intelligent Systems and Technology*, 7(1). doi:[10.1145/2743026](https://doi.org/10.1145/2743026)
- Fu, W.-T., & Dong, W. (2012). Collaborative indexing and knowledge exploration: A social learning model. *IEEE Intelligent Systems*, 27(1), 39–46. doi:[10.1109/MIS.2010.131](https://doi.org/10.1109/MIS.2010.131)
- Fu, W.-T., Kannampallil, T., Kang, R., & He, J. (2010). Semantic imitation in social tagging. *ACM Transactions on Computer-Human Interaction*, 17(3). doi:[10.1145/1806923.1806926](https://doi.org/10.1145/1806923.1806926)
- Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11), 964–971. doi:[10.1145/32206.32212](https://doi.org/10.1145/32206.32212)
- Golder, S. A., & Huberman, B. A. (2006). The structure of collaborative tagging systems. *Journal of Information Science*, 32(2), 198–208. doi:[10.1177/0165551506062337](https://doi.org/10.1177/0165551506062337)
- Hintzman, D. L. (1986). "Schema abstraction" in a multiple trace-model. *Psychological Review*, 93(4), 411–428. doi:[10.1037/0033-295X.93.4.411](https://doi.org/10.1037/0033-295X.93.4.411)
- Hintzman, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review*, 95, 528–551. doi:[10.1037/0033-295X.95.4.528](https://doi.org/10.1037/0033-295X.95.4.528)
- Hutchins, E., & Johnson, C. (2009). Modeling the emergence of language as an embodied collective cognitive activity. *Trends in Cognitive Science*, 13(3), 523–546. doi:[10.1111/j.1756-8765.2009.01033.x](https://doi.org/10.1111/j.1756-8765.2009.01033.x)
- Janis, I. L. (1972). *Victims of groupthink: A psychological study of foreign-policy decisions and fiascoes*. Boston, MA: Houghton Mifflin.
- Jäschke, R., Marinho, L., Hotho, A., Schmidt-Thieme, L., & Stumme, G. (2007). Tag recommendations in folksonomies. In *Knowledge discovery in databases: PKDD 2007. Lecture notes in computer science* (pp. 506–514). Berlin, Heidelberg: Springer. doi:[10.1007/978-3-540-74976-9_52](https://doi.org/10.1007/978-3-540-74976-9_52)
- Kaplan, I. T., Carvellas, T., & Metlay, W. (1969). Searching for words in letter sets of varying sizes. *Journal of Experimental Psychology*, 82(2), 377–380. doi:[10.1037/h0028140](https://doi.org/10.1037/h0028140)
- Kerne, A., Smith, S., Koh, E., Choi, H., & Graeber, R. (2008). An experimental method for measuring the emergence of new ideas in information discovery. *International Journal of Human-Computer Interaction*, 24(5), 460–477. doi:[10.1080/10447310802142243](https://doi.org/10.1080/10447310802142243)
- Kerne, A., Webb, A., Smith, S., Linder, R., Lupfer, N., Qu, Y., ... Damaraju, S. (2014). Using metrics of curation to evaluate information-based ideation. *ACM Transactions on Computer-Human Interaction*, 21(3), 14:1–14:48. doi:[10.1145/2591677](https://doi.org/10.1145/2591677)
- Kohn, N., Paulus, P., & Choi, Y. (2011). Building on the ideas of others: An examination of the idea combination process. *Journal of Experimental Social Psychology*, 47, 554–561. doi:[10.1016/j.jesp.2011.01.004](https://doi.org/10.1016/j.jesp.2011.01.004)
- Kowald, D., Seitlinger, P., Kopeinik, S., Ley, T., Albert, D., & Trattner, C. (2014). Forgetting the words but remembering the meaning: Modeling forgetting in a verbal and semantic tag recommender. In M. Atzmueller, A. Chin, C. Scholz, & C. Trattner (Eds.), *Mining, modeling, and recommending things in social media* (pp. 75–95). Heidelberg, Germany: Springer Lecture Notes in Artificial Intelligence. doi:[10.1007/978-3-319-14723-9_5](https://doi.org/10.1007/978-3-319-14723-9_5)
- Kwanten, P. J. (2005). Using context to build semantics. *Psychonomic Bulletin & Review*, 12(4), 703–710. doi:[10.3758/BF03196761](https://doi.org/10.3758/BF03196761)
- Latour, B. (2005). *Reassembling the social. An introduction to actor-network theory*. Oxford, New York: Oxford University Press.

- Lazer, D., & Bernstein, E. S. (2012). Problem solving and search in networks. In P. M. Todd, T. T. Hills, & T. W. Robbins (Eds.), *Cognitive search* (pp. 269–282). Cambridge, MA: MIT Press.
- Lazer, D., & Friedman, A. (2007). The network structure of exploration and exploitation. *Administrative Science Quarterly*, 52, 667–694. doi:[10.2189/asqu.52.4.667](https://doi.org/10.2189/asqu.52.4.667)
- Linder, R., Snodgrass, C., & Kerne, A. (2014). Everyday ideation: All of my ideas are on interest. In A. Schmidt & T. Grossman (Eds.), *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 2411–2420). New York, NY: ACM press. doi:[10.1145/2556288.2557273](https://doi.org/10.1145/2556288.2557273)
- Lorince, J., & Todd, P. (2016). Music tagging and listening: Testing the memory cue hypothesis in a collaborative tagging system. In M. Jones (Ed.), *Big data in cognitive science*. New York, NY: Psychology Press.
- Martindale, C. (1995). Creativity and connectionism. In S. Smith, T. Ward, & R. Finke (Eds.), *The creative cognition approach* (pp. 249–268). Cambridge, MA: MIT Press.
- Mednick, S. (1962). The associative basis of the creative process. *Psychological Review*, 69(3), 220–232. doi:[10.1037/h0048850](https://doi.org/10.1037/h0048850)
- Nelson, L., Held, C., Pirolli, P., Hong, L., Schiano, D., & Chi, E. H. (2009). With a little help from my friends: Examining the impact of social annotations in sensemaking tasks. In K. Hinckley, M. Ringel, S. Hudson, & S. Greenberg (Eds.), *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 1795–1798). New York, NY: ACM press. doi:[10.1145/1518701.1518977](https://doi.org/10.1145/1518701.1518977)
- Nijstad, B., & Stroebe, W. (2006). How the group affects the mind: A cognitive model of idea generation in groups. *Personality and Social Psychology Review*, 10(3), 186–213. doi:[10.1207/s15327957pspr1003_1](https://doi.org/10.1207/s15327957pspr1003_1)
- Page, S. E. (2007). *The difference: How the power of diversity creates better groups, firms, schools, and societies*. Princeton, NJ: Princeton University Press.
- Pariser, E. (2011). *The filter bubble: What the internet is hiding from you*. New York, NY: Penguin Press.
- Paulus, P., & Brown, V. (2007). Towards more creative and innovative group idea generation: A cognitive-social-motivational perspective of brainstorming. *Social and Personality Psychology Compass*, 1(1), 248–265. doi:[10.1111/j.1751-9004.2007.00006.x](https://doi.org/10.1111/j.1751-9004.2007.00006.x)
- Pirolli, P., & Kairam, S. (2012). A knowledge-tracing model of learning from a social tagging system. *User Modeling and User-Adapted Interaction*, 23(2), 139–168. doi:[10.1007/s11257-012-9132-1](https://doi.org/10.1007/s11257-012-9132-1)
- Polyn, S. M., Norman, K. A., & Kahana, M. J. (2009). A context maintenance and retrieval model of organizational processes in free recall. *Psychological Review*, 116(1), 129–156. doi:[10.1037/a0014420](https://doi.org/10.1037/a0014420)
- Raijmakers, J. G. W., & Shiffrin, R. M. (1980). SAM: A theory of probabilistic search of associative memory. In G. H. Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory* (Vol 14, pp. 207–262). New York, NY: Academic Press.
- Renner, B., Prilla, M., Cress, U., & Kimmerle, J. (2016). Effects of prompting in reflective learning tools: findings from experimental field, lab, and online studies. *Frontiers in Psychology*, 7. <http://doi.org/10.3389/fpsyg.2016.00820>
- Roepstorff, A., Niewöhner, J., & Beck, S. (2010). Enculturing brains through patterned practices. *Neural Networks*, 23(8–9), 1051–1059. doi:[10.1016/j.neunet.2010.08.002](https://doi.org/10.1016/j.neunet.2010.08.002)
- Rohrer, D. (2002). The breadth of memory search. *Memory*, 10(4), 291–301. doi:[10.1080/09658210143000407](https://doi.org/10.1080/09658210143000407)
- Sarmiento, J. W., & Stahl, G. (2008). Group creativity in interaction: Collaborative referencing, remembering, and bridging. *International Journal of Human-Computer Interaction*, 24(5), 492–504. doi:[10.1080/10447310802142300](https://doi.org/10.1080/10447310802142300)
- Schweiger, S., Oeberst, A., & Cress, U. (2014). Confirmation bias in web-based search: A randomized online study on the effects of expert information and social tags on information search and evaluation. *Journal of Medical Internet Research*, 16(3), e94. doi:[10.2196/jmir.3044](https://doi.org/10.2196/jmir.3044)
- Seitlinger, P., & Ley, T. (2012). Implicit imitation in social tagging: Familiarity and semantic reconstruction. In E. H. Chi & K. Höök (eds), *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 1631–1640). New York, NY: ACM press. doi:[10.1145/2207676.2208287](https://doi.org/10.1145/2207676.2208287)
- Seitlinger, P., & Ley, T. (2016). Reconceptualizing imitation in social tagging: A reflective search model of human web interaction. In P. Parigi & S. Staab (Eds.), *Proceedings of the 8th ACM conference on web science* (pp. 146–155). New York, NY: ACM press. doi:[10.1145/2908131.2908157](https://doi.org/10.1145/2908131.2908157)
- Seitlinger, P., Ley, T., & Albert, D. (2013). An implicit-semantic tag recommendation mechanism for socio-cognitive learning systems. In T. Ley, M. Ruohonen, M. Laanpere, & A. Tatnall (Eds.), *Proceedings of Open and Social Technologies for networked learning OST'12* (pp. 41–46). Heidelberg, Germany Springer LNCS. doi:[10.1007/978-3-642-37285-8_5](https://doi.org/10.1007/978-3-642-37285-8_5)
- Seitlinger, P., Ley, T., & Albert, D. (2015). Verbatim and semantic imitation in indexing resources on the Web: A fuzzy-trace account of social tagging. *Applied Cognitive Psychology*, 29(1), 32–48. doi:[10.1002/acp.3067](https://doi.org/10.1002/acp.3067)
- Smith, S. M., Ward, T. B., & Finke, R. A. (1995). *The creative cognition approach*. Cambridge, MA: MIT Press.
- Sprenger, A., Dougherty, M., Atkins, S., Franco-Watkins, A., Thomas, R., Lange, N., & Abbs, B. (2011). Implications of cognitive load for hypothesis generation and probability judgment. *Frontiers in Psychology*, 2. doi:[10.3389/fpsyg.2011.00129](https://doi.org/10.3389/fpsyg.2011.00129)
- Sunstein, C. (2001). *Echo chambers: Bush v. Gore, Impeachment, and Beyond*. Princeton, Oxford: Princeton University Press.
- Trattner, C., Kowald, D., Seitlinger, P., Kopeinik, S., & Ley, T. (2016). Modeling activation processes in human memory to predict the use of tags in social bookmarking systems. *The Journal of Webscience*, 2(1), 1–16. doi:[10.1561/106.00000004](https://doi.org/10.1561/106.00000004)
- Unsworth, N., & Engle, R. W. (2007). The nature of individual differences in working memory capacity: Active maintenance in primary memory and controlled search from secondary memory. *Psychological Review*, 114(1), 104–132. doi:[10.1037/0033-295X.114.1.104](https://doi.org/10.1037/0033-295X.114.1.104)
- Wagner, C., Singer, P., Strohmaier, M., & Huberman, B. (2014). Semantic stability in social tagging streams. In A. Broder, K. Shim, & T. Suel (Eds.), *Proceedings of the 23rd international conference on world wide web* (pp. 735–746). New York, NY: ACM Press. doi:[10.1145/2566486.2567979](https://doi.org/10.1145/2566486.2567979)
- Ward, T. B. (2007). Creative cognition as a window on creativity. *Methods*, 42(1), 28–37. doi:[10.1016/j.jymeth.2006.12.002](https://doi.org/10.1016/j.jymeth.2006.12.002)
- Webster, J., Gibbins, N., Halford, S., & Hracs, B. (2016). Towards a theoretical approach for analyzing music recommender systems as sociotechnical cultural intermediaries. In P. Parigi & S. Staab (Eds.), *Proceedings of the 8th ACM Conference on Web Science* (pp. 137–145). New York, NY: ACM press. doi:[10.1145/2908131.2908148](https://doi.org/10.1145/2908131.2908148)
- Wixted, J., & Rohrer, D. (1994). Analyzing the dynamics of free recall: An integrative review of the empirical literature. *Psychonomic Bulletin & Review*, 1(1), 10–89. doi:[10.3758/BF03200763](https://doi.org/10.3758/BF03200763)

About the Authors

Dr Paul Seitlinger is a senior researcher at the School of Educational Sciences (Tallinn University). His main interest is the role of human memory in reflective search on the Web and in the design of ideationally stimulating information services. Paul has a PhD in Psychology from the University of Graz.

Dr Tobias Ley is Professor of Learning Analytics and Educational Innovation (Tallinn University). He has published over 100 publications in leading outlets in HCI and technology-enhanced learning. He has had leading roles in a number of large-scale EU-funded projects. Tobias has a PhD in Psychology from the University of Graz.

Dr Dominik Kowald is university assistant at the Institute of Interactive Systems and Data Science (Graz University of Technology). Additionally, he is senior researcher in the Social Computing research area of Know-Center Graz. He has recently finished his PHD thesis on cognitive-inspired recommender systems for social tagging environments

Dipl.-Ing. Dieter Theiler is software engineer at Know-Center Graz (Social Computing research area). He studied Software Engineering and Economy at Graz University of Technology and was responsible for the software engineering process in the course of the EU-IP Learning Layers. His interests include software architecture and engineering, TEL and CSCW.

Dipl.-Ing. Ilire Hasani-Mavriqi is PhD student at the Institute of Interactive Systems and Data Science (Graz University of Technology) and part of the Social Computing team at Know-Center Graz. Her research interests include social network analysis, dynamics in online networks, opinion diffusion and dynamics of social influence in complex networks.

Mag. Sebastian Dennerlein is deputy head of the Social Computing research area and business developer at Know-Center Graz. He studied Psychology at the University of Graz and is doing his PhD

on Collaboration and Meaning Making in socio-technical systems. His research interests are cognitive psychology, TEL, CSCL, and CSCW.

Dr Elisabeth Lex, Asst.-Professor at Graz University of Technology, heads the Social Computing research area at Know-Center Graz. Her research interests include Dynamics in Complex Networks, Recommender Systems, Web Science, Open Science and Machine Learning. She is also member of the Expert Group on Altmetrics, which advises the European Commission.

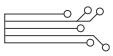
Dr Dietrich Albert, Univ.-Professor, is (a) Senior Scientist at the *Graz University of Technology* (b) Prof. em. at the *University of Graz*, and (c) Key Researcher at the *Know-Center*. Combining *Cognitive Science* and *Information Communication Technology* in different fields is the focus of his current research, development, and innovation (<http://css-kmi.tugraz.at/research/projectvis/tagcloud.html>).

P2 Modeling Popularity and Temporal Drift of Music Genre Preferences (2020)

Transparency and Cognitive Models in Recommender Systems

[P2] Lex, E.*, **Kowald, D.***, Schedl, M. (2020). Modeling Popularity and Temporal Drift of Music Genre Preferences. *Transactions of the International Society for Music Information Retrieval*, 3:1, pp. 17-30. (*equal contribution)

DOI: <https://doi.org/10.5334/tismir.39>



RESEARCH

Modeling Popularity and Temporal Drift of Music Genre Preferences

Elisabeth Lex^{*†}, Dominik Kowald^{*‡} and Markus Schedl[§]

In this paper, we address the problem of modeling and predicting the music genre preferences of users. We introduce a novel user modeling approach, BLL_u , which takes into account the popularity of music genres as well as temporal drifts of user listening behavior. To model these two factors, BLL_u adopts a psychological model that describes how humans access information in their memory. We evaluate our approach on a standard dataset of Last.fm listening histories, which contains fine-grained music genre information. To investigate performance for different types of users, we assign each user a mainstreamness value that corresponds to the distance between the user's music genre preferences and the music genre preferences of the (Last.fm) mainstream. We adopt BLL_u to model the listening habits and to predict the music genre preferences of three user groups: listeners of (i) niche, low-mainstream music, (ii) mainstream music, and (iii) medium-mainstream music that lies in-between. Our results show that BLL_u provides the highest accuracy for predicting music genre preferences, compared to five baselines: (i) group-based modeling, (ii) user-based collaborative filtering, (iii) item-based collaborative filtering, (iv) frequency-based modeling, and (v) recency-based modeling. Besides, we achieve the most substantial accuracy improvements for the low-mainstream group. We believe that our findings provide valuable insights into the design of music recommender systems.

Keywords: Music Genre Preference Prediction; Music Recommendation; Music Retrieval; Personalized Music Access; Time-Aware Recommendation; ACT-R

Publisher's Note

The corresponding author was changed to Elisabeth Lex and the statement referring to the TU Graz Open Access Publishing Fund was added on 14/04/2020.

1. Introduction

Music recommender systems play a pivotal role in popular streaming platforms such as Last.fm,¹ Pandora,² or Spotify³ to help users find music that suits their taste. Existing music recommender systems typically employ collaborative filtering algorithms based on the users' interactions with music items (i.e., listening behavior or ratings), sometimes in combination with content features (e.g., acoustic features of songs) in the form of hybrid music recommender systems (Celma, 2010; Schedl et al., 2018b).

Problem. While music recommender systems can provide quality recommendations to listeners of popular music, related research (Schedl and Bauer, 2018; van den Oord et al., 2013) has shown that they tend to fail listeners who prefer niche artists and genres. A reason for that is the

scarcity of usage data of such types of music as music consumption patterns are biased towards popular artists (van den Oord et al., 2013; Celma, 2010; Celma and Cano, 2008). In this paper, we introduce a novel user modeling and genre prediction approach for users with different music consumption patterns and listening habits. We focus on three user groups: (i) LowMS, i.e., listeners of niche music, (ii) HighMS, i.e., listeners of mainstream (MS) music, and (iii) MedMS, i.e., listeners of music that lies in-between. The main problem we address in this work is how to exploit variations in listening habits to improve personalization for all three user groups. We investigate this problem by predicting the music genres a user is going to listen to in the future.

Approach and methods. We model the users' listening behavior in terms of fine-grained music genre preferences. To that end, we use behavioral data in the form of listening events, i.e., the listening history of which genres a user has listened to in the past. Our approach is based on the Base-Level Learning (BLL) equation from the cognitive architecture ACT-R (Anderson and Schooler 1991; Anderson et al., 2004) that accounts for the time-dependent decay of item exposure in human memory. It quantifies the usefulness of a piece of information based on how frequently and recently a user accessed it in the past. This time-dependent decay takes the shape of a power-law distribution. Related work has employed the

^{*} Both authors contributed equally to this work

[†] Graz University of Technology, Graz, AT

[‡] Know-Center GmbH, Graz, AT

[§] Johannes Kepler University (JKU) Linz and Linz Institute of Technology (LIT) AI Lab, Linz, AT

Corresponding author: Dominik Kowald (dkowald@know-center.at)

BLL equation to recommend Web links (Fu and Pirolli, 2007), to recommend scientific talks at conferences (Maanen and Marewski, 2009), to recommend tags in social bookmarking systems (Kowald and Lex, 2016), and to recommend hashtags (Kowald et al., 2017b).

In this work, we build upon these results and adopt the BLL equation to model the listening habits of users in our three groups to predict their music genre preferences. We demonstrate the efficacy of our approach on the *LFM-1b* dataset (Schedl, 2016), which contains listening histories of more than 120,000 Last.fm users, amounting to 1.1 billion individual listening events over nine years. The music in this dataset is categorized according to a fine-grained taxonomy that consists of 1,998 music genres and styles. Additionally, the dataset contains demographic data such as age and gender as well as a “mainstreaminess” factor (Bauer and Schedl, 2019) that relates the listening preferences of each user to the aggregated preferences of all Last.fm users in the dataset. Based on this factor, we assign the users in our dataset to one of the three groups, i.e., (i) LowMS, (ii) MedMS, and (iii) HighMS. This allows us to evaluate our proposed BLL_u approach for different types of users.

Contributions and findings. The contributions of our work are two-fold. Firstly, we propose the BLL_u approach for modeling popularity and temporal drift of music genre preferences. Secondly, we evaluate BLL_u on three different groups of Last.fm users, which we separate based on the distance of their listening behavior to the mainstream: (i) LowMS, (ii) MedMS, and (iii) HighMS.

We find that for all three groups, BLL_u provides the highest accuracy for predicting music genre preference, compared to five baselines: (i) group-based modeling (i.e., *TOP*), (ii) user-based collaborative filtering (i.e., CF_u), (iii) item-based collaborative filtering (i.e., CF_i), (iv) frequency-based modeling (i.e., POP_u), and (v) recency-based modeling (i.e., $TIME_u$). Moreover, BLL_u gives the highest accuracy improvements for the LowMS group. Finally, we also validate our findings in a cold-start setting, in which we only evaluate users with a small number of listening events. Here, we also find that our BLL_u approach provides the best prediction accuracy results.

Structure of this paper. This paper is organized as follows: In Section 2, we review related work, and in Section 3, we describe the dataset as well as statistical analyses about genre mainstreamness, popularity, and temporal drift of music genre preferences. Also, this section includes the methodology and the proposed approach for modeling music genre preferences. In Section 4, we present the experimental setup as well as the evaluation results. Finally, Section 5 concludes this paper and gives an outlook into future work.

2. Related Work

At present, we identify three strands of related research: (i) research on music preferences in light of psychology, (ii) temporal dynamics of music preferences, and (iii) personalization for music recommendation.

Research on music preferences in light of psychology.

Research in music psychology (North and Hargreaves, 2008) has shown that a range of factors impact music preferences (Schedl et al., 2015), such as emotional state (Cantor and Zillmann, 1973; Juslin and Sloboda, 2001; Rodà et al., 2014), a user’s current activity, their self-view and self-esteem (North and Hargreaves, 1999), the cognitive functions of music (e.g., music as a way to communicate and to self-reflect) (Schäfer and Sedlmeier, 2010), as well as personality (Cattell and Anderson, 1953; Arnett, 1992; Dollinger, 1993; Rentfrow and Gosling, 2003; George et al., 2007; Delsing et al., 2008; Dunn et al., 2012; Schedl et al., 2018a).

For instance, Rentfrow and Gosling (2003) showed that the Big Five personality traits (i.e., openness to experience, agreeableness, extraversion, neuroticism, and conscientiousness) influence genre preferences in music and that music preferences can be categorized along specific dimensions (e.g., reflective & complex, intense & rebellious, upbeat & conventional, and energetic & rhythmic music); the structure of music preferences is also discussed by Delsing et al. (2008). Greenberg et al. (2015) found that a person’s cognitive approach (i.e., their tendency towards empathy versus systemizing versus balancing both) impacts their music genre preferences. A user’s music preference is also impacted by familiarity (Pereira et al., 2011; Schubert, 2007). This has been attributed to the so-called *mere exposure effect* (Peretz et al., 1998), which means that prior exposure can positively influence music liking. In our work, we also incorporate prior exposure (in this case, to a music genre) into our model.

Temporal dynamics of music preferences. Music preferences are often dynamic due to variations in user taste (Kim et al., 2018), or evolving music taste (Moore et al., 2013). One can distinguish between research on long-term temporal dynamics of listening behavior and short-term dynamics. Studies investigating long-term dynamics research on, for example, how music preferences of children and young adults evolve (Hargreaves et al., 2015; Leadbeater, 2014), or how user tastes change over time and how artists develop (Moore et al., 2013).

Studies investigating short-term dynamics typically assess users’ listening behaviors (Aizenberg et al., 2012; Park and Kahng, 2010) on a fine-granular basis (e.g., time of the day) to detect patterns and periodicity in listening behavior, or in the case of Krause and North (2018), to study the relationship between music preferences and seasons of the year. The latter approaches are typically intended to help create predictive models of music preferences to create playlist recommendations for music streaming services, among others. As we describe in detail in Section 3, in our data, we observe interesting temporal dynamics in users’ genre listening histories. Specifically, the time-dependent decay of number of plays per genre follows a power-law distribution, so our users tend to listen to genres to which they have recently listened.

Personalization for music recommendation. A number of aspects make personalization in music recommender

systems challenging, such as, e.g., the variability of listening intent and purpose of music consumption, insufficient ratings and usage data, as well as users' tendency to appreciate recommendations of items that have been previously recommended (Schedl et al., 2018b), but also the dependence of music preferences on the user's personality traits or emotional state. In this vein, Selvi and Sivasankar (2019) extracted the user's emotional context from social media messages as well as their current time context and incorporated both to generate personalized music recommendations. Ferwerda et al. (2015) used a specific personality-enriched dataset that provided links to users' listening histories on Last.fm to leverage personality traits to predict a user's genre preferences. Zheng et al. (2018) proposed a tag-aware dynamic music recommendation framework that represents musical tracks via user-generated tags and generates time-sensitive recommendations. Koenigstein et al. (2011) incorporated a temporal analysis of user ratings assigned to music pieces and item popularity trends into a matrix factorization approach to mitigate the issue of insufficient item ratings. The latter is a common problem that causes (music) recommender systems to suffer from bias towards popular items. Due to insufficient amounts of usage data for less popular items, many recommendation algorithms cannot provide useful recommendations for consumers of less popular and niche items (Abdollahpouri et al., 2019; Celma, 2010; van den Oord et al., 2013). Recent work (Vall et al., 2019) has yet provided evidence that deep-learning-based methods (i.e., recurrent neural networks) seem to be less biased towards popular items.

In our work, we use only listening histories as a data source to model user preferences and to generate recommendations. As we show in Section 3, we observe that all users in our dataset tend to consume items they have listened to frequently and recently in the past, where the time-dependent decay of this item consumption count follows a power-law distribution. Correspondingly, the Base-Level Learning (BLL) equation from the cognitive architecture ACT-R (Anderson and Schooler, 1991; Anderson et al., 2004) describes a time-dependent decay of item exposure in human memory in the form of a power-law distribution. Leveraging these similarities between characteristics of music consumption patterns and cognition models (i.e., ACT-R in our case), we propose here to use the BLL equation to describe listeners' behavioral music consumption traces.

3. Data and Method

In this section, we present the dataset we use for our study and statistical analyses we carry out. We outline the approach of this work and the baselines, which we employ to validate our proposed method.

3.1 Dataset and Statistical Analyses

First, we describe the Last.fm dataset, as well as the selected genre mapping procedure. We report statistical analyses for (i) music genre popularity, (ii) average pairwise user similarity, (iii) popularity of music genre preferences, and (iv) temporal drifts of music genre preferences.

Dataset description and availability. For our study, we use a dataset gathered from the online music service Last.fm, namely the *LFM-1b* dataset.⁴ *LFM-1b* contains listening histories of more than 120,000 users, totaling to about 1.1 billion individual listening events accrued between January 2005 and August 2014. Each listening event is characterized by a user identifier, artist, album, track name, and a timestamp (Schedl 2016). Besides, the *LFM-1b* dataset contains user-specific demographic data such as country, age, gender as well as additional features such as mainstreaminess, which is defined as the overlap between the user's listening history and the aggregated listening history of all Last.fm users in the dataset. More precisely, the mainstreaminess of a user corresponds to the average distance between all artists' relative frequencies in the user's listening profile and the artists' relative frequencies among all users in the dataset (Schedl and Hauger, 2015).

Mapping listening events to music genres. Since we are interested in modeling and predicting music genre preferences, we enhance the listening events in the *LFM-1b* dataset with additional genre information. Therefore, we use an extension of the *LFM-1b* dataset, termed LFM-1b User-Genre-Profile (i.e., *LFM-1b UGP*) dataset (Schedl and Ferwerda, 2017), which describes the genres of an artist in a listening event by exploiting social tags from Last.fm.

Among others, *LFM-1b UGP* contains a weighted mapping of 1,998 music genres and styles available in the online database Freebase⁵ to Last.fm artists. In part, this taxonomy includes particular descriptors such as "Progressive Psytrance" or "Melodic Black Metal", and therefore allows for a fine-grained representation of musical styles. The weightings correspond to the relative frequency of tags assigned to artists in Last.fm. For example, for the artist "Metallica" the top tags and their corresponding relative frequencies are "thrash metal" (.10), "metal" (.91), "heavy metal" (.74), "hard rock" (.41), "rock" (.34) and "seen live" (.3). This means that the tag "thrash metal" is the most popular genre tag assigned to "Metallica" and thus, its weighting is 1.0. From this list, we remove all tags that are not part of the 1,998 Freebase genres (i.e., "seen live" in our example) as well as all tags with a relative frequency smaller than .5 (i.e., "hard rock" and "rock" in our example). Thus, for "Metallica", we end up with three genres, namely "thrash metal", "metal" and "heavy metal" that we assign to all listening events of the artist "Metallica". Overall, this process gives us, on average, 2–3 genres per artist (i.e., mean = 2.466). Furthermore, 96.25% of the genres are assigned to more than one artist.

User groups based on mainstreaminess. The *LFM-1b* dataset contains a mainstreaminess value for each user, which defines the distance from this user's music genre preferences to the music genre preferences of the (Last.fm) mainstream. To study different types of users, we split the dataset into three equally sized groups based on their mainstreaminess (i.e., low, medium, and high). We sort the users in the dataset based on their mainstreaminess value and assign the 1,000 users with the lowest values to the

LowMS group, the 1,000 users with the highest values to the HighMS group, and the 1,000 users with a value that lies around the average mainstreaminess ($=.379$) to the MedMS group.

Here, we consider only users with at least 6,000 and at most 12,000 listening events, a choice we made based on the average number of listening events per user in the dataset (i.e., 9,043) as well as the kernel density distribution of the data. With this method, on the one hand, we exclude users with too little data available for training our algorithms (i.e., users with $<6,000$ listening events), and on the other hand, we exclude so-called power listeners (i.e., users with $>12,000$ listening events) who might distort our results.

Furthermore, this high average number of listening events per user also means that we have enough listening events (i.e., between 6.9 to 8.2 million) to train and test the music genre preference modeling and prediction approaches, even if we only consider 1,000 users per group. **Table 1** summarizes the statistics and characteristics of these three groups.

(i) LowMS. The LowMS group represents the $|U| = 1,000$ least mainstream users. They have an average mainstreaminess value of $\overline{MS} = .125$. This group contains $|A| = 82,417$ distinct artists, $|LE| = 6,915,352$ listening events, $|G| = 931$ genres and $|GA| = 14,573,028$ genre assignments.

(ii) MedMS. The MedMS group represents the $|U| = 1,000$ users whose mainstreamness values are between the ones of LowMS and HighMS groups (i.e., their mainstreamness values lie around the average). This group has an average mainstreaminess value of $\overline{MS} = .379$. Most statistics of this group lie between those of the LowMS and HighMS users (for example, the number of genre assignments per listening event $|GA|/|LE| = 2.565$), except for the average age, which is the highest for the MedMS users ($\overline{Age} = 25.352$ years).

(iii) HighMS. This group represents the $|U| = 1,000$ most mainstream users in the *LFM-1b* dataset ($\overline{MS} = .688$). These users are not only the youngest ones ($\overline{Age} = 21.486$ years) but also listen to the highest number of distinct genres on average ($\overline{G_u} = 186.010$). Also, this user group exhibits the highest number of distinct genres ($|G| = 973$).

Average pairwise user similarity. Finally, the boxplots in **Figure 1** show the average pairwise user similarity in the three user groups. We calculate these scores based on

Table 1: Dataset statistics for the LowMS, MedMS, and HighMS Last.fm user groups. Here, $|U|$ is the number of distinct users, $|A|$ is the number of distinct artists, $|G|$ is the number of distinct genres, $|LE|$ is the number of listening events, $|GA|$ is the number of genre assignments, $|GA|/|LE|$ is the number of genre assignments per listening event, $\overline{G_u}$ is the average number of genres a user u has listened to, \overline{MS} is the average mainstreaminess value, and \overline{Age} is the average age of users in the group.

User Group	$ U $	$ A $	$ G $	$ LE $	$ GA $	$ GA / LE $	$\overline{G_u}$	\overline{MS}	\overline{Age}
LowMS	1,000	82,417	931	6,915,352	14,573,028	2.107	85.771	.125	24.582
MedMS	1,000	86,249	933	7,900,726	20,264,870	2.565	126.439	.379	25.352
HighMS	1,000	92,690	973	8,251,022	22,498,370	2.727	186.010	.688	21.486

the genre distributions of the users and using the cosine similarity metric. We see that users in the LowMS group have a very individual listening behavior (mean user similarity = .118), while users in the HighMS group tend to listen to similar music genres (mean user similarity = .691). Again, the users in the MedMS group lie in between (mean user similarity = .392). Given these results, we expect a collaborative filtering approach based on user similarities to deliver good genre prediction results for the HighMS group.

Popularity of music genre preferences. In **Figure 2**, we compare the music genre popularity distributions of the LowMS, MedMS, and HighMS groups. To this end, we plot the number of listening events for the groups' top-30 genres. We find that there are some dominating genres with more than 2 million LE counts in the HighMS group, while the genre distribution is much more evenly distributed in the LowMS group with a LE count of around 500,000 for the most popular genres. We can describe the genre distribution of the MedMS group as an intermediate of the LowMS and HighMS distribution. We analyze the actual top-30 genres in these groups, and while the most popular genres Rock and Pop dominate the other genres

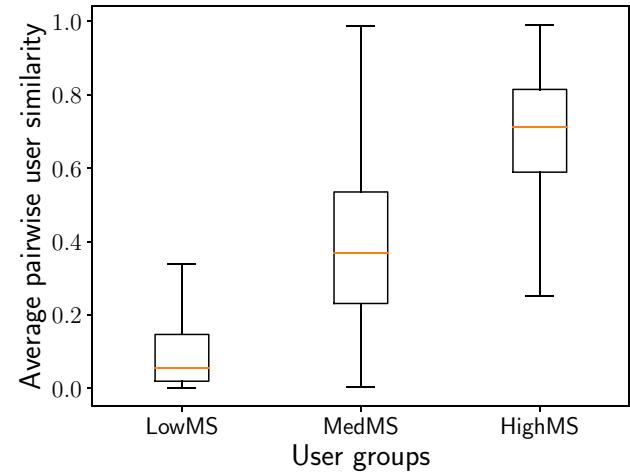


Figure 1: Boxplots show the average pairwise user similarity in our user groups using the cosine similarity metric computed on the users' genre distributions. While users in the LowMS group show a very individual listening behavior, users in the HighMS group tend to listen to similar music genres.

in the HighMS group (LE count of Rock = 2,269,861), in the LowMS group, it is not as dominant (LE count of Rock = 685,998). Furthermore, we find several genres that are not popular in the MedMS and HighMS groups but are popular in the LowMS group, such as Ambient and Black Metal.

Based on the dataset characteristics, we expect that a group-based modeling approach, which models a user's music genre preferences utilizing the most-frequently listened genres of all users in the group, performs fine for HighMS in relation to other modeling techniques, while for the LowMS group, a personalized modeling technique would be preferable. In the MedMS group, we expect both modeling approaches to work well due to the group being an intermediate of the HighMS and LowMS groups.

Temporal drift of music genre preferences. Next, we investigate the temporal drift of music genre preferences. The plots (a), (b), and (c) of **Figure 3** show the effect of time on the genre listening behavior of our LowMS, MedMS, and HighMS user groups. We plot the relistening

count of music genres over the time (in hours) since the last listening events of these genres on a log-log scale. For example, if a user u has listened to artists with genre g twice in a time interval of 1 hour, then the relistening count for "1 hour" is incremented by 1. We repeat this process for all listening events, which gives us a relistening count for each hour. We observe similar results for all three groups, which means that the shorter the time since the last listening event of a genre g , the higher its relistening count. In all three plots, we see a peak after 24 hours, which indicates that people tend to listen to similar music genres daily at the same time. However, we also see that when people have not listened to a genre for a longer period, i.e., one month (around 750 hours), the relistening count of this genre drastically drops.

Finally, we also plot the linear regression lines of the empirical data in the plots of **Figure 3**. In the log-log-scaled plots, we can observe a good fit of the data, which indicates that the data likely follows a power-law distribution (cf. Anderson and Schooler, 1991). This claim is supported by the high R^2 values of the fits, which are between .870 and .895. Concerning the slopes α of the lines, which describe how strongly temporal listening drifts influence the user groups, we observe values between -1.480 and -1.587 . We can use these values as the d parameter of the BLL equation (Anderson et al., 2004), cf. Equation 6.

Taken together, we observe interesting temporal effects in all three user groups: Last.fm users tend to listen to genres they have listened to recently. Moreover, we find that this temporal drift of music genre preferences follows a power-law distribution. Correspondingly, we can model this drift with the BLL equation.

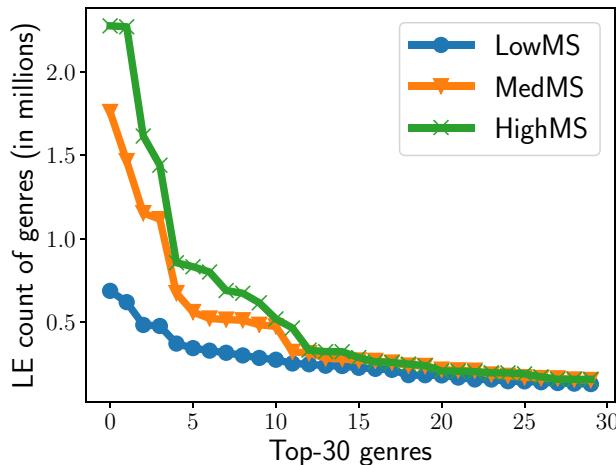


Figure 2: Number of listening events LE (in millions) for the top-30 genres of our LowMS, MedMS, and HighMS Last.fm user groups. We find that there are some dominating genres in the HighMS group, while the genre distribution in the LowMS group is more evenly distributed.

3.2 Modeling and Prediction of Music Genre Preferences

In this section, we describe five baseline approaches (i.e., TOP , CF_u , CF_p , POP_u , and $TIME_u$) as well as our approach based on the BLL equation for modeling and predicting music genre preferences (i.e., BLL_u).

Group-based baseline: TOP . Motivated by our analysis in **Figure 2**, the TOP approach models a user u 's music genre preferences using the overall top- k (e.g., top-30) genres

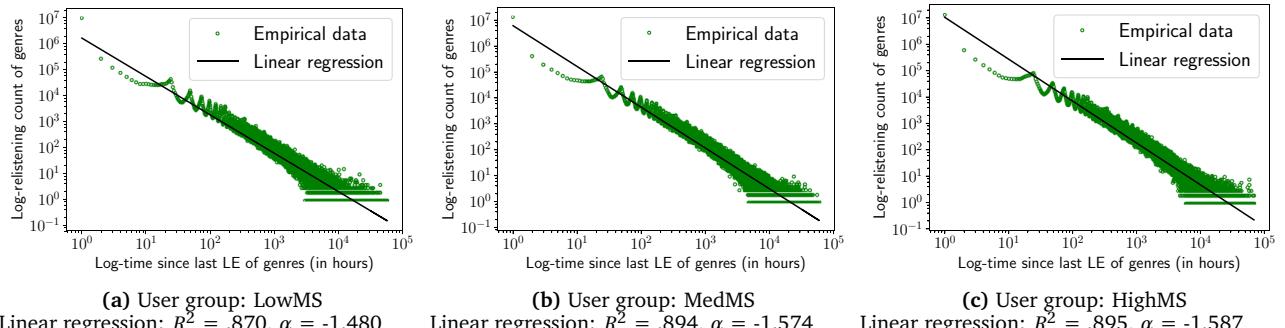


Figure 3: The effect of time on genre relistening behavior for the LowMS, MedMS, and HighMS Last.fm user groups. For all three groups, we find that the shorter the time since the last listening event of a genre, the higher its relistening count. Additionally, we plot the linear fits of the data and report the corresponding R^2 estimates as well as the slopes α . We can observe a very good fit of the data, which indicates that the data likely follows a power-law distribution.

of all users in the user group UG_u (i.e., LowMS, MedMS, HighMS) to which u belongs. This is given by:

$$\widetilde{G}_u^k = \underset{g \in G}{\operatorname{argmax}}(|GA_{g,UG_u}|) \quad (1)$$

where argmax^k refers to the “arguments of the maxima” function for the top- k genres with maximum values, \widetilde{G}_u^k denotes the set of k predicted genres for user u , and $|GA_{g,UG_u}|$ corresponds to the number of times g occurs in all genre assignments GA of UG_u . Thus, we describe this approach as a group-based modeling technique since it reflects the preferences of the whole user group LowMS, MedMS or HighMS. As our analysis in **Figure 2** shows that the genre distribution in the HighMS group is the least evenly distributed one, we expect the *TOP* approach to provide good prediction accuracy results for the HighMS group while performing worse for the LowMS group in relation to other modeling techniques.

User-based collaborative filtering baseline: CF_u . User-based collaborative filtering-based approaches aim to find similar users for a target user u , i.e., the set of neighbors N_u . N_u is calculated using the cosine similarity between u 's genre distribution and the genre distributions of all other users. Then, the top-20 users are defined as N_u . Finally, CF_u predicts the genres these similar users in N_u have listened to (Shi et al., 2014), which is formally given by:

$$\widetilde{G}_u^k = \underset{g \in G}{\operatorname{argmax}} \left(\sum_{v \in N_u} sim(G_u, G_v) \cdot |GA_{g,v}| \right) \quad (2)$$

where $sim(G_u, G_v)$ is the cosine similarity between the genre distributions of user u and neighbor v , and $|GA_{g,v}|$ indicates how often v has listened to genre g . Since CF_u relies on user similarities, we expect it to provide good results for the HighMS group compared to other modeling approaches (see also **Figure 1**).

Item-based collaborative filtering baseline: CF_i . Similar to CF_u , CF_i is a collaborative filtering-based approach, but instead of finding similar users for the target user u , it aims to find similar items (i.e., music artists). Then it predicts the genres that are assigned to these similar artists as given by:

$$\widetilde{G}_u^k = \underset{g \in G}{\operatorname{argmax}} \left(\sum_{a \in A_u} \sum_{s \in S_a} sim(G_a, G_s) \cdot |GA_{g,s}| \right) \quad (3)$$

Here, A_u is the set of artists u has listened to, S_a is the set of similar artists for an artist a , $sim(G_a, G_s)$ is the cosine similarity between the genres assigned to a and the genres assigned to a similar artist s , and $|GA_{g,s}|$ indicates how often genre g was assigned to artist a (hence, in our case either 0 or 1). Again, a neighborhood size $|S_{A_u}| = 20$ leads to the best genre prediction results, and we also set A_u to the set of the 20 artists that u has listened to most frequently.

Frequency-based baseline: POP_u . The POP_u approach is a personalized music genre preference modeling technique, which predicts the k most frequently listened to (i.e., most popular) genres in the listening history of a user u . POP_u corresponds to the modeling approach presented in (Schedl and Ferwerda, 2017) and is given by the following equation:

$$\widetilde{G}_u^k = \underset{g \in G_u}{\operatorname{argmax}}(|GA_{g,u}|) \quad (4)$$

where G_u is the set of genres u has listened to⁶ and $|GA_{g,u}|$ denotes the number of times u has listened to tracks with genre g (i.e., the frequency). Thus, it ranks the genres u has listened to in the past by popularity. Therefore, in relation to other modeling algorithms, we expect POP_u to generate good genre predictions for all users in our three user groups, but especially for HighMS, in which the popularity feature is the most important one (see **Figure 2**).

Recency-based baseline: $TIME_u$. Our analysis presented in **Figure 3** motivates the personalized and recency-based music genre preference modeling, where we find that people tend to listen to genres to which they have listened just very recently. Thus, $TIME_u$ predicts the most recently listened to genres that are present in the listening history of a user u , which is given by:

$$\widetilde{G}_u^k = \underset{g \in G_u}{\operatorname{argmin}}(t_{u,g,n}) \quad (5)$$

where $t_{u,g,n}$ is the time since the last (i.e., the n^{th}) listening event of g by u . Since we find that the temporal drift of music genre preferences is an important feature for all our three user groups, $TIME_u$ should provide good prediction accuracy results for LowMS, MedMS, and HighMS in relation to other modeling approaches.

Our approach based on the BLL equation: BLL_u . To combine the frequency-based modeling method POP_u with the recency-based modeling method $TIME_u$, we utilize the BLL equation from the declarative memory module of the cognitive architecture ACT-R (Anderson et al., 2004). The BLL equation quantifies the importance of information in human memory (e.g., a word or a music genre) by considering how recently (i.e., temporal drift) and frequently (i.e., popularity) it was used in the past. In our setting, we define it as follows:

$$B_{u,g} = \ln \left(\sum_{j=1}^n t_{u,g,j}^{-d} \right) \quad (6)$$

Here, g is a genre user u has listened to in the past, and n is the number of times u has listened to g . Further, $t_{u,g,j}$ is the time since the j^{th} listening event of g by u , and d is the power-law decay factor that accounts for the feature of the temporal drift of music genre preferences.

We set d to the slopes α identified in the analysis of **Figure 3** (i.e., 1.480 for LowMS, 1.574 for MedMS, and

1.587 for HighMS). The resulting base-level activation values $B'_{u,g}$ are normalized using a simple softmax function in order to map them onto a range of [0,1] where they sum to 1 (Kowald et al., 2017b):

$$B'_{u,g} = \frac{\exp(B_{u,g})}{\sum_{g' \in G_u} \exp(B_{u,g'})} \quad (7)$$

Again, G_u is the set of distinct genres listened to by u . Finally, BLL_u predicts the top- k genres \tilde{G}_u^k with the highest $B'_{u,g}$ values for u :

$$\tilde{G}_u^k = \operatorname{argmax}_{g \in G_u} (B'_{u,g}) \quad (8)$$

Comparison of approaches. **Table 2** shows how the five baselines, as well as BLL_u , cover our four features of interest, i.e., (i) personalization, (ii) collaboration, (iii) popularity, and (iv) temporal drift.

Here, our BLL_u approach is the only one that covers the features of personalization, popularity, and temporal drifts. Moreover, TOP , CF_u , and CF_i are the only approaches that consider collaboration among users and, thus, investigate the listening events of all users. We further examine which feature combination works best for predicting genres in our setting in the next section of this paper.

4. Experiments and Results

In this section, we outline the experimental setup (see Section 4.1) and in Section 4.2, we present the results of our study on evaluating the usefulness for modeling music genre preferences using the BLL equation.

4.1 Experimental Setup

To measure the accuracy of our music genre preference modeling approaches, we conduct a study, in which we predict the genres assigned to the artists a user is going to listen to in the future.

Evaluation protocol. We split the datasets into train and test sets (Cremonesi et al., 2008) and make sure that our

Table 2: Comparison of our five baselines as well as our approach based on the BLL equation for modeling and predicting music genre preferences. In this table, a “✓” indicates that a specific approach covers a specific feature. While TOP , CF_u and CF_i also consider collaboration among users (i.e., investigate listening events of all users), our BLL_u approach is the only one that is personalized and accounts for the features of popularity as well as temporal drifts.

Feature	TOP	CF_u	CF_i	POP_u	$TIME_u$	BLL_u
Personalization	✓	✓	✓	✓	✓	✓
Collaboration	✓	✓	✓			
Popularity	✓	✓	✓	✓		✓
Temporal drifts				✓	✓	

evaluation protocol preserves the temporal order of the listening events, which simulates a real-world scenario in which we predict (genres of) future listening events based on past ones (Kowald et al., 2017b; Seitlinger et al., 2015). This also means that a classic k -fold cross-validation evaluation protocol with random splits is not useful.

Therefore, we put the most recent 1% of the listening events of each user into the test set and keep the remaining listening events for training. We do not use a classic 80/20 or 90/10 split as the number of listening events per user is large (i.e., on average 7,689 per user). Furthermore, although we only use the most recent 1% of listening events per user, this process leads to three large test sets with 69,153 listening events for LowMS, 79,007 listening events for MedMS, and 82,510 listening events for HighMS. On average, there are 76 listening events per user for which we predict the assigned genres.

In **Figure 4**, we present boxplots showing the average duration in days per user we have available in our three test sets. We see that the average duration per user is evenly distributed across all three user groups with a median value of 11.8 days, which is also around 1% of the median value of the overall average duration per user (i.e., the sum of training and test durations). This corresponds to the 1% of the listening events per user we use for the test sets. Thus, we are going to predict the genres a user is going to listen to in this period.

Following this evaluation protocol, our goal is to validate whether our BLL-based approach (i.e., BLL_u) provides better prediction accuracy results than the five baseline approaches (i.e., TOP , CF_u , CF_i , POP_u , and $TIME_u$). When investigating the numbers shown in **Table 1**, we also see that our prediction task is not trivial since $|GA|/|LE|$, i.e., the number of genre assignments per listening event (=what should be predicted), is much smaller than \bar{G}_u , i.e., the average number of genres a user u has listened to (=what could be predicted).

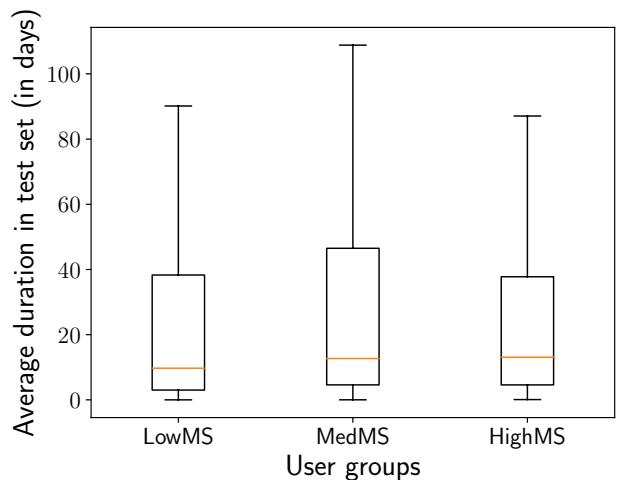


Figure 4: Boxplots showing the average duration in days per user we have available in our three test sets. Across all three users groups, the average duration per user is evenly distributed with a median value of 11.8 days.

Evaluation metrics. To measure the prediction quality of the approaches, we use the following six state-of-the-art metrics (Baeza-Yates and Ribeiro-Neto, 2011):

(i) **Recall:** $R@k$. Recall is calculated as the number of correctly predicted genres divided by the number of relevant genres (i.e., from the test set). It is a measure of the completeness of the predictions.

(ii) **Precision:** $P@k$. Precision is calculated as the number of correctly predicted genres divided by the number of predictions k and is a measure of the accuracy of the predictions. We report recall and precision for $k = 1 \dots 10$ predicted genres in the form of recall/precision plots.

(iii) **F1-score:** $F1@5$. F1-score is the harmonic mean of recall and precision. If 10 genres are predicted, the F1-score typically reaches its highest value for $k = 5$. Thus, we report it for $k = 5$.

(iv) **Mean Reciprocal Rank:** $MRR@10$. MRR is the mean of reciprocal ranks of all relevant genres in the list of predicted genres.

(v) **Mean Average Precision:** $MAP@10$. MAP is the mean of the average precision scores at all ranks where relevant genres are predicted. With this, it also takes the ranking of the correctly predicted genres into account.

(vi) **Normalized Discounted Cumulative Gain:** $nDCG@10$. nDCG is another ranking-dependent metric. It is based on the Discounted Cumulative Gain (DCG) measure (Järvelin et al., 2008).

We report MRR, MAP, and nDCG for $k = 10$ predicted music genres, where these metrics reach their highest values.

Evaluation framework. For reasons of reproducibility, we conduct the prediction study using our recommendation benchmarking framework *TagRec* (Kowald et al., 2017a), which provides the evaluation protocol and metrics

described in this section. Furthermore, we also implement the modeling approaches described in Section 3.2 using *TagRec*. It is freely available via our Github repository.⁷

4.2 Results and Discussion

In this section, we report and discuss our prediction accuracy results on evaluating the usefulness of our BLL_u -based music genre preference modeling approach (i.e., BLL_u) compared to five baseline approaches: (i) group-based modeling (i.e., TOP), (ii) user-based collaborative filtering (CF_u), (iii) item-based collaborative filtering (CF_i), (iv) frequency-based modeling (i.e., POP_u), and (v) recency-based modeling (i.e., $TIME_u$).

Table 3 summarizes our evaluation results for the three user groups (i.e., LowMS, MedMS, and HighMS), the four evaluation metrics (i.e., $F1@5$, $MRR@10$, $MAP@10$, and $nDCG@10$) as well as the six approaches (i.e., TOP , CF_u , CF_i , POP_u , $TIME_u$, and BLL_u). Additionally, in **Figure 5**, we show the recall/precision plots of the approaches for $k = 1 \dots 10$ predicted genres (i.e., $R@k$ and $P@k$).

Based on the features introduced in **Table 2**, we discuss these results concerning the influence of (i) personalization, (ii) collaboration, (iii) popularity, and (iv) temporal drift. Furthermore, we compare the results of our BLL_u approach for our user groups and different numbers of predicted genres in **Figure 6** as well as show the performance of the approaches in a cold-start setting in **Figure 7**. Finally, we also discuss the implications of our findings for personalized music recommendation.

Influence of personalization. The personalized approaches (i.e., POP_u , CF_u , CF_i , $TIME_u$, and BLL_u) outperform the group-based TOP approach in the LowMS setting. This is in line with our analysis presented in **Figure 2**, where we

Table 3: Genre prediction accuracy results of our study comparing our BLL_u approach with a group-based baseline (TOP), a user-based collaborative filtering baseline (CF_u), an item-based collaborative filtering baseline (CF_i), a frequency-based baseline (POP_u) and a recency-based baseline ($TIME_u$). For all three user groups (i.e., LowMS, MedMS, and HighMS), the combination of popularity and temporal drift of music genre preferences in the form of BLL_u provides the best results for all metrics. According to a t-test with $\alpha = .001$, “***” indicates statistically significant differences between BLL_u and all other approaches for all user groups.

User group	Evaluation metric	TOP	CF_u	CF_i	POP_u	$TIME_u$	BLL_u
LowMS	$F1@5$.108	.311	.341	.356	.368	.397***
	$MRR@10$.101	.389	.425	.443	.445	.492***
	$MAP@10$.112	.461	.505	.533	.550	.601***
	$nDCG@10$.180	.541	.590	.618	.625	.679***
MedMS	$F1@5$.196	.271	.284	.292	.293	.338***
	$MRR@10$.146	.248	.264	.274	.272	.320***
	$MAP@10$.187	.319	.336	.351	.365	.419***
	$nDCG@10$.277	.419	.441	.460	.452	.523***
HighMS	$F1@5$.247	.273	.266	.282	.228	.304***
	$MRR@10$.188	.232	.229	.242	.201	.266***
	$MAP@10$.246	.304	.298	.314	.267	.348***
	$nDCG@10$.354	.413	.402	.429	.357	.462***

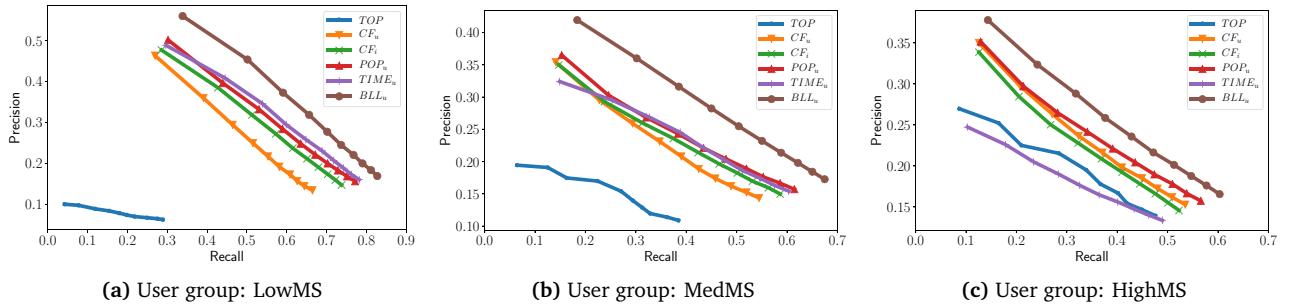


Figure 5: Recall/precision plots of the baselines and our BLL_u approach for the three user groups LowMS, MedMS, and HighMS. We see that BLL_u provides the best results for all groups and for all $k = 1 \dots 10$ predicted genres.

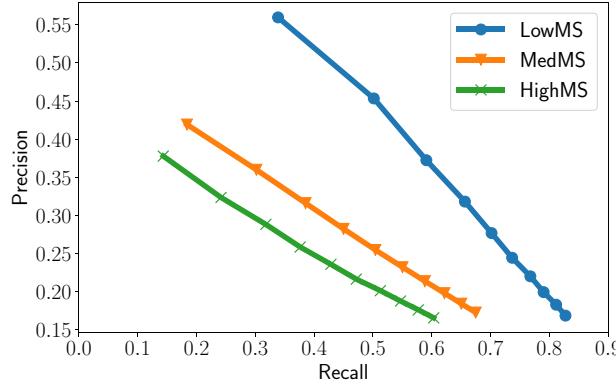


Figure 6: Recall/precision plot of our BLL_u approach for $k = 1 \dots 10$ predicted genres for the three user groups LowMS, MedMS and HighMS. We see that BLL_u provides good prediction accuracy results for all groups but especially in the LowMS setting. This shows that our approach is especially useful for predicting the music genre preferences of users with low mainstreamness values.

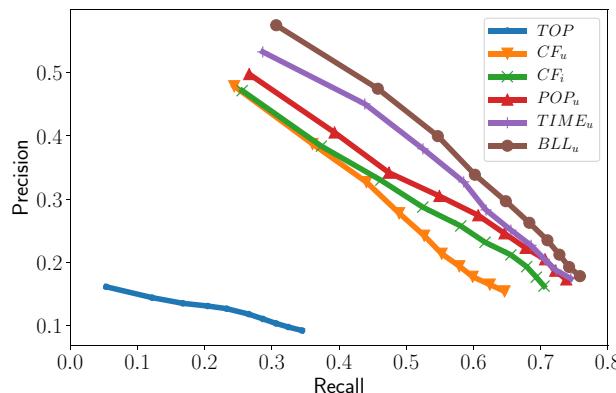


Figure 7: Recall/precision plot for our BLL_u approach and our five baselines in a cold-start setting. We see that BLL_u also provides the best results in cases where users only have a few listening events available for training.

found that the music genre popularity distribution in the LowMS group is the most evenly distributed one.

The same is true for the MedMS group, in which we observe a very similar performance of CF_u , CF_i , POP_u , and $TIME_u$. However, in the HighMS setting only the four personalized approaches, which utilize the popularity feature (i.e., POP_u , CF_u , CF_i , and BLL_u) outperform TOP . This shows that the influence of personalization on the

prediction accuracy becomes more important as the mainstreamness of the users decreases (i.e., in the LowMS setting).

Influence of collaboration. We investigate the genre prediction accuracy of three approaches (i.e., TOP , CF_u and CF_i) that consider collaboration among users, i.e., that analyze the listening events of all users. Here, the personalized CF_u and CF_i approaches provide better results than the non-personalized TOP approach for all three user groups.

Furthermore, CF_u provides its best results for the HighMS group. This is in line with our analysis presented in **Figure 1**, which shows that the average pairwise user similarity is the highest for high-mainstream users. This is also the reason why CF_i does not outperform CF_u in the HighMS but outperforms it in the LowMS and MedMS settings.

Influence of popularity. We evaluate four popularity-based approaches. The first approach provides non-personalized genre predictions based on the preferences of all users (i.e., TOP), and the second offers personalized predictions based on user similarities (i.e., CF_u). The third approach provides personalized predictions using item similarities (i.e., CF_i), and the fourth produces personalized genre predictions based on the preferences of the individual user (i.e., POP_u). While the prediction accuracy of TOP increases with the level of mainstreamness, the prediction accuracy of POP_u decreases with the level of mainstreamness. The prediction accuracy of CF_u and CF_i are relatively stable over all three user groups, with the only exception that CF_u provides better results than CF_i in the HighMS setting.

Thus, in the HighMS group, TOP provides a higher prediction accuracy than in the other two groups. These results are in line with our analysis presented in **Figure 2**, where we find that there are some dominating genres in the HighMS group, which explains the good results of TOP , CF_u and POP_u in this setting. When further comparing CF_u with CF_i we see that CF_i outperforms CF_u in the LowMS and MedMS settings.

Influence of temporal drift. Our analysis in **Figure 3** reveals that users in Last.fm tend to listen to genres which they have listened to very recently. In other words, time is

important for all three user groups. However, as shown in **Table 3** and **Figure 5**, $TIME_u$ provides the weakest accuracy results for HighMS and good prediction accuracy results for LowMS and MedMS. Thus, for HighMS, popularity is a more important feature than recency.

BLL_u outperforms $TIME_u$ in all experiments. This means that our personalized modeling approach, which also considers the features of popularity and temporal drifts, can provide accurate genre predictions for all three groups in relation to other modeling techniques.

Accuracy of BLL_u for different values of k . In **Figure 6**, we show the recall/precision results of BLL_u for $k = 1 \dots 10$ predicted genres for the three user groups. We observe apparent differences in the accuracy value ranges when comparing the three groups. While BLL_u outperforms the five baselines in all three settings (with significant differences between BLL_u and all other approaches according to a t-test with $\alpha = .001$), the accuracy estimates are much higher in the LowMS group (i.e., $R@10 = .827$ and $P@1 = .559$) than in the MedMS group (i.e., $R@10 = .674$ and $P@1 = .419$) and the HighMS group (i.e., $R@10 = .603$ and $P@1 = .377$). This shows that our approach is especially useful to predict the genre preferences of users with low inclination to listen to mainstream music.

Performance in cold-start setting. Since recommender systems are often faced with situations in which users only have a few interactions available to train the underlying recommendation algorithms, we also evaluate our BLL_u approach in a cold-start setting (Schein et al., 2002). For this, we extract the 1,000 users with the lowest number of LEs from the LFM-1b dataset. As we need to make sure that we have at least 1 LE per user available for training the algorithms, this procedure leads to 1,000 users with a minimum of 2 LEs and a maximum of 46 LEs per user. For these users, we have precisely 1 LE in the test set, for which we predict the assigned genres.

Our results for this experiment are shown in the recall/precision plot of **Figure 7**. Here, we observe very similar results to the ones of our LowMS, MedMS, and HighMS settings (see **Figure 6**). Thus, again BLL_u provides the best accuracy results followed by $TIME_u$, POP_u , CF_p , and CF_u . As expected, the non-personalized TOP approach provides the worst results in this setting. These results show that BLL_u is also capable of effectively predicting music genre preferences in cold-start settings where users only have a few listening events available for training.

Implications for personalized music recommendation. In this section, so far, we have shown that BLL_u outperforms the baseline approaches concerning prediction accuracy in different settings (i.e., LowMS, MedMS, HighMS, and cold-start). When looking at **Figure 6**, this is especially true for the LowMS group, in which users do not follow the preferences of the mainstream, and thus, a personalization technique, as given by the BLL equation, is critical. If we relate this to music recommender systems, which exploit the listening histories of users to suggest

other music that they might also like, our findings lead to interesting implications. Schedl and Hauger (2015) have shown that standard recommendation algorithms such as collaborative filtering cannot provide suitable music recommendations for users with low mainstreamness. The results presented in this section support this. In other words, such users need different music recommendation algorithms that account for their highly individual listening preferences.

One way to achieve this could be to combine state-of-the-art music recommendation algorithms (see Section 2) with our music genre preference modeling approach based on the BLL equation presented in this paper. We could use the calculated B'_{ug} values given by our approach as an input for these algorithms or to rerank recommendation results based on the importance of a genre for a user. We elaborate on these ideas as well as other plans for future work in Section 5.

5. Conclusion and Future Work

In this paper, we presented BLL_u , an approach that utilizes the features of popularity and temporal drifts to model and predict music genre preferences via fine-grained genres. We leveraged the LFM-1b dataset of more than one billion music listening events, created by approximately 120,000 users of the online music service Last.fm. We divided the users into three groups based on the proximity of their music genre preferences to the mainstream: (i) LowMS, i.e., listeners of niche music, (ii) HighMS, i.e., listeners of mainstream music, and (iii) MedMS, i.e., listeners of music that lies in-between. To take into account the popularity and temporal drift of music genre preferences, we proposed to use the Base-Level Learning (BLL) equation from the cognitive architecture ACT-R, which quantifies the importance of information in human memory (e.g., a music genre) by considering how frequently (i.e., popularity) and recently (i.e., temporal drift) it was used in the past. A comparison between BLL_u and a group-based baseline (i.e., TOP), a user-based collaborative filtering baseline (i.e., CF_u), an item-based collaborative filtering baseline (i.e., CF_p), a frequency-based baseline (i.e., POP_u) as well as a recency-based baseline (i.e., $TIME_u$) showed that BLL_u outperforms all other approaches for all three user groups in terms of prediction accuracy.

Furthermore, our results indicate that BLL_u is especially useful to predict the music genre preferences of users with interest in low-mainstream music (i.e., the LowMS user group), which opens up interesting possibilities for future work in the research area of personalized music recommender systems.

Limitations and future work. So far, we limited our approach to the BLL equation of the declarative memory module of ACT-R. Since the BLL equation is only a part of the more exhaustive ACT-R framework that does not consider contextual information, one needs to consider this limitation when utilizing our approach. For example, when we model music genre preferences exclusively via past listening behavior, phenomena such as over-personalization or filter-bubble effects could occur

(Nguyen et al., 2014). To overcome this, we plan to extend our model to the full activation equation of ACT-R, which also considers contextual information via its associative activation (Anderson et al., 2004). Moreover, we plan to extend our model by other components of ACT-R, for example, to investigate further context dimensions such as the mood or the current activity of the user (see, e.g., Ferwerda et al. (2015)). We could achieve this by defining and implementing so-called production rules from ACT-R's procedural memory module as, for instance, done in the SNIF-ACT model (Pirolli and Fu, 2003; Fu and Pirolli, 2007). Another limitation of our work is that we employed a rather simple definition for the mainstreaminess of a user. We, therefore, plan to extend our analysis to include more sophisticated mainstreaminess measures, e.g., based on rank-order correlation or Kullback-Leibler divergence (Schedl and Bauer, 2018). As part of future work, we plan to integrate our findings into music recommendation algorithms, with particular attention to addressing the low mainstreaminess group, since standard collaborative filtering approaches tend to fail to provide suitable music recommendations for this user group (Schedl and Hauger, 2015). For example, we plan to integrate the preference values we obtain for a specific user and a particular genre via our approach as a context dimension into a matrix factorization-based approach (Mnih and Salakhutdinov, 2008; Koenigstein et al., 2011) or a deep learning-based approach (Lin et al., 2018; Sachdeva et al., 2018).

Furthermore, we aim to apply our approach to the problem of music playlist continuation, which was also the task of the ACM RecSys Challenge 2018.⁸ We believe that our findings concerning the temporal relistening patterns of music genres (see Section 3.1) could help identify genres that users commonly listened to consecutively. We could then, for example, incorporate such genre sequences into the two-stage convolutional neural network (CNN) model for automatic playlist continuation that was proposed by Volkovs et al. (2018). Finally, we would like to highlight that our approach could be easily leveraged by researchers and practitioners also for other related tasks (e.g., recommending music artists) and not only for genre prediction. Thus, we hope that future work in the areas of user modeling and music recommendation will be attracted by our insights.

Reproducibility

To foster the reproducibility of our research, we use the publicly available LFM-1b Last.fm dataset (see Section 3.1). Furthermore, we provide our evaluation framework *TagRec* (see Section 4.1) freely for academic purposes. We hope that the approach presented in this paper and its implementation in *TagRec*, as well as the dataset, will attract further research on music preference modeling and recommender systems.

Notes

¹ <https://www.last.fm/>.

² <https://www.pandora.com/>.

³ <https://www.spotify.com/>.

⁴ <http://www.cp.jku.at/datasets/LFM-1b/>.

⁵ <https://developers.google.com/freebase/> (no longer maintained).

⁶ Here, we could also use G instead of G_u , which would lead to the same results, but to reduce the computational effort, we only need to consider the genres that the target user u has listened to in the past.

⁷ <https://github.com/learning-layers/TagRec>.

⁸ <http://www.recsyschallenge.com/2018/>.

Acknowledgements

We thank Peter Muellner for his valuable feedback on this work. This work was supported by the H2020 projects AI4EU and TRIPLE, and the Know-Center GmbH. The Know-Center GmbH is funded within the Austrian COMET (Competence Centers for Excellent Technologies) Program under the auspices of the Austrian Ministry of Transport, Innovation and Technology, the Austrian Ministry of Economics and Labor and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency (FFG).

Competing Interests

The authors have no competing interests to declare.

Author Contributions

Elisabeth Lex and Dominik Kowald contributed equally to this work.

References

- Abdollahpouri, H., Mansouri, M., Burke, R., & Mobasher, B.** (2019). The unfairness of popularity bias in recommendation. *arXiv preprint arXiv:1907.13286*.
- Aizenberg, N., Koren, Y., & Somekh, O.** (2012). Build your own music recommender by modeling internet radio streams. In *Proceedings of the International World Wide Web Conference*, pages 1–10. ACM. DOI: <https://doi.org/10.1145/2187836.2187838>
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y.** (2004). An integrated theory of the mind. *Psychological Review*, 111(4). DOI: <https://doi.org/10.1037/0033-295X.111.4.1036>
- Anderson, J. R., & Schooler, L. J.** (1991). Reflections of the environment in memory. *Psychological Science*, 2(6), 396–408. DOI: <https://doi.org/10.1111/j.1467-9280.1991.tb00174.x>
- Arnett, J.** (1992). The soundtrack of recklessness: Musical preferences and reckless behavior among adolescents. *Journal of Adolescent Research*, 7(3), 313–331. DOI: <https://doi.org/10.1177/074355489273003>
- Baeza-Yates, R., & Ribeiro-Neto, B.** (2011). *Modern Information Retrieval*. ACM Press. DOI: <https://doi.org/10.1145/2009916.2010172>
- Bauer, C., & Schedl, M.** (2019). Global and country-specific mainstreaminess measures: Definitions, analysis, and usage for improving personalized music recommendation systems. *PLoS ONE*, 14(6), 1–36. DOI: <https://doi.org/10.1371/journal.pone.0217389>
- Cantor, J. R., & Zillmann, D.** (1973). The effect of affective state and emotional arousal on music appreciation. *The Journal of General Psychology*, 89(1), 97–108. DOI: <https://doi.org/10.1080/00221309.1973.9710822>

- Cattell, R. B., & Anderson, J. C.** (1953). The measurement of personality and behavior disorders by the IPAT music preference test. *Journal of Applied Psychology*, 37(6), 446. DOI: <https://doi.org/10.1037/h0056224>
- Celma, O.** (2010). *Music Recommendation and Discovery – The Long Tail, Long Fail, and Long Play in the Digital Music Space*. Springer. DOI: <https://doi.org/10.1007/978-3-642-13287-2>
- Celma, Ò., & Cano, P.** (2008). From hits to niches?: Or how popular artists can bias music recommendation and discovery. In *Proceedings of the 2nd Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*. ACM. DOI: <https://doi.org/10.1145/1722149.1722154>
- Cremonesi, P., Turrin, R., Lentini, E., & Matteucci, M.** (2008). An evaluation methodology for collaborative recommender systems. In *Proceedings of International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution*, pages 224–231. IEEE Computer Society. DOI: <https://doi.org/10.1109/AXMEDIS.2008.13>
- Delsing, M. J., Ter Bogt, T. F., Engels, R. C., & Meeus, W. H.** (2008). Adolescents' music preferences and personality characteristics. *European Journal of Personality: Published for the European Association of Personality Psychology*, 22(2), 109–130. DOI: <https://doi.org/10.1002/per.665>
- Dollinger, S. J.** (1993). Research note: Personality and music preference: Extraversion and excitement seeking or openness to experience? *Psychology of Music*, 21(1), 73–77. DOI: <https://doi.org/10.1177/030573569302100105>
- Dunn, P. G., de Ruyter, B., & Bouwhuis, D. G.** (2012). Toward a better understanding of the relation between music preference, listening behavior, and personality. *Psychology of Music*, 40(4), 411–428. DOI: <https://doi.org/10.1177/0305735610388897>
- Ferwerda, B., Yang, E., Schedl, M., & Tkalcic, M.** (2015). Personality traits predict music taxonomy preferences. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*, pages 2241–2246. ACM. DOI: <https://doi.org/10.1145/2702613.2732754>
- Fu, W.-T., & Pirolli, P.** (2007). SNIF-ACT: A cognitive model of user navigation on the World Wide Web. *Human-Computer Interaction*, 22(4), 355–412. DOI: <https://doi.org/10.21236/ADA462156>
- George, D., Stickle, K., Rachid, F., & Wopnford, A.** (2007). The association between types of music enjoyed and cognitive, behavioral, and personality factors of those who listen. *Psychomusicology: A Journal of Research in Music Cognition*, 19(2). DOI: <https://doi.org/10.1037/h0094035>
- Greenberg, D. M., Baron-Cohen, S., Stillwell, D. J., Kosinski, M., & Rentfrow, P. J.** (2015). Musical preferences are linked to cognitive styles. *PLoS ONE*, 10(7), 1–22. DOI: <https://doi.org/10.1371/journal.pone.0131151>
- Hargreaves, D. J., North, A. C., & Tarrant, M.** (2015). How and why do musical preferences change in childhood and adolescence. *The Child as Musician: A Handbook of Musical Development*, pages 303–322. DOI: <https://doi.org/10.1093/acprof:oso/9780198744443.003.0016>
- Järvelin, K., Price, S. L., Delcambre, L. M., & Nielsen, M. L.** (2008). Discounted cumulated gain based evaluation of multiple-query IR sessions. In *Proceedings of the European Conference on Information Retrieval*, pages 4–15. Springer. DOI: https://doi.org/10.1007/978-3-540-78646-7_4
- Juslin, P. N., & Sloboda, J. A.** (2001). *Music and Emotion: Theory and Research*. Oxford University Press.
- Kim, N., Chae, W.-Y., & Lee, Y.-J.** (2018). Music recommendation with temporal dynamics in multiple types of user feedback. In *Proceedings of the 7th International Conference on Emerging Databases*, pages 319–328. Springer. DOI: https://doi.org/10.1007/978-981-10-6520-0_35
- Koenigstein, N., Dror, G., & Koren, Y.** (2011). Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item taxonomy. In *Proceedings of ACM Conference on Recommender Systems*, pages 165–172. ACM. DOI: <https://doi.org/10.1145/2043932.2043964>
- Kowald, D., Kopeinik, S., & Lex, E.** (2017a). The TagRec framework as a toolkit for the development of tag-based recommender systems. In *Adjunct Publication of the ACM Conference on User Modeling, Adaption and Personalization*, pages 23–28. ACM. DOI: <https://doi.org/10.1145/3099023.3099069>
- Kowald, D., & Lex, E.** (2016). The influence of frequency, recency and semantic context on the reuse of tags in social tagging systems. In *Proceedings of ACM Conference on Hypertext and Social Media*, pages 237–242. ACM. DOI: <https://doi.org/10.1145/2914586.2914617>
- Kowald, D., Pujari, S. C., & Lex, E.** (2017b). Temporal effects on hashtag reuse in twitter: A cognitiveinspired hashtag recommendation approach. In *Proceedings of the International World Wide Web Conference*, pages 1401–1410. ACM. DOI: <https://doi.org/10.1145/3038912.3052605>
- Krause, A. E., & North, A. C.** (2018). 'Tis the season: Music-playlist preferences for the seasons. *Psychology of Aesthetics, Creativity, and the Arts*, 12(1). DOI: <https://doi.org/10.1037/aca0000104>
- Leadbeater, R.** (2014). *Magpies and mirrors: identity as a mediator of music preferences across the lifespan*. PhD thesis, Lancaster University.
- Lin, Q., Niu, Y., Zhu, Y., Lu, H., Mushonga, K. Z., & Niu, Z.** (2018). Heterogeneous knowledge-based attentive neural networks for short-term music recommendations. *IEEE Access*, 6. DOI: <https://doi.org/10.1109/ACCESS.2018.2874959>
- Maanen, L. V., & Marewski, J. N.** (2009). Recommender systems for literature selection: A competition between decision making and memory models. In *Proceedings of the Annual Meeting of the Cognitive Science Society*.
- Mnih, A., & Salakhutdinov, R. R.** (2008). Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, pages 1257–1264.

- Moore, J. L., Chen, S., Turnbull, D., & Joachims, T.** (2013). Taste over time: The temporal dynamics of user preferences. In *Proceedings of the International Society for Music Information Retrieval Conference*, pages 401–406.
- Nguyen, T. T., Hui, P.-M., Harper, F. M., Terveen, L., & Konstan, J. A.** (2014). Exploring the filter bubble: The effect of using recommender systems on content diversity. In *Proceedings of the International World Wide Web Conference*, pages 677–686. ACM. DOI: <https://doi.org/10.1145/2566486.2568012>
- North, A., & Hargreaves, D.** (2008). *The Social and Applied Psychology of Music*. OUP Oxford. DOI: <https://doi.org/10.1093/acprof:oso/9780198567424.001.0001>
- North, A. C., & Hargreaves, D. J.** (1999). Music and adolescent identity. *Music Education Research*, 1(1), 75–92. DOI: <https://doi.org/10.1080/1461380990010107>
- Park, C. H., & Kahng, M.** (2010). Temporal dynamics in music listening behavior: A case study of online music service. In *Proceedings of the IEEE/ACIS International Conference on Computer and Information Science*, pages 573–578. IEEE. DOI: <https://doi.org/10.1109/ICIS.2010.142>
- Pereira, C. S., Teixeira, J., Figueiredo, P., Xavier, J., Castro, S. L., & Brattico, E.** (2011). Music and emotions in the brain: Familiarity matters. *PLoS ONE*, 6(11). DOI: <https://doi.org/10.1371/journal.pone.0027241>
- Peretz, I., Gaudreau, D., & Bonnel, A.-M.** (1998). Exposure effects on music preference and recognition. *Memory & Cognition*, 26(5), 884–902. DOI: <https://doi.org/10.3758/BF03201171>
- Pirolli, P., & Fu, W.-T.** (2003). SNIF-ACT: A model of information foraging on the World Wide Web. In *International Conference on User Modeling*, pages 45–54. Springer. DOI: https://doi.org/10.1007/3-540-44963-9_8
- Rentfrow, P. J., & Gosling, S. D.** (2003). The do re mi's of everyday life: The structure and personality correlates of music preferences. *Journal of Personality and Social Psychology*, 84(6). DOI: <https://doi.org/10.1037/0022-3514.84.6.1236>
- Rodà, A., Canazza, S., & Poli, G. D.** (2014). Clustering affective qualities of classical music: Beyond the valence-arousal plane. *IEEE Transactions on Affective Computing*, 5(4), 364–376. DOI: <https://doi.org/10.1109/TAFFC.2014.2343222>
- Sachdeva, N., Gupta, K., & Pudi, V.** (2018). Attentive neural architecture incorporating song features for music recommendation. In *Proceedings of the ACM Conference on Recommender Systems*, pages 417–421. ACM. DOI: <https://doi.org/10.1145/3240323.3240397>
- Schäfer, T., & Sedlmeier, P.** (2010). What makes us like music? Determinants of music preference. *Psychology of Aesthetics, Creativity, and the Arts*, 4(4). DOI: <https://doi.org/10.1037/a0018374>
- Schedl, M.** (2016). The lfm-1b dataset for music retrieval and recommendation. In *Proceedings of the Conference on Multimedia Retrieval*, pages 103–110. ACM. DOI: <https://doi.org/10.1145/2911996.2912004>
- Schedl, M., & Bauer, C.** (2018). An analysis of global and regional mainstreamness for personalized music recommender systems. *Journal of Mobile Multimedia*, 14, 95–112.
- Schedl, M., & Ferwerda, B.** (2017). Large-scale analysis of group-specific music genre taste from collaborative tags. In *Proceedings of the IEEE International Symposium on Multimedia*, pages 479–482. IEEE. DOI: <https://doi.org/10.1109/ISM.2017.95>
- Schedl, M., Gómez, E., Trent, E., Tkalcic, M., Eghbal-Zadeh, H., & Martorell, A.** (2018a). On the interrelation between listener characteristics and the perception of emotions in classical orchestra music. *IEEE Transactions on Affective Computing*, 9, 507–525. DOI: <https://doi.org/10.1109/TAFFC.2017.2663421>
- Schedl, M., & Hauger, D.** (2015). Tailoring music recommendations to users by considering diversity, mainstreamness, and novelty. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 947–950. ACM. DOI: <https://doi.org/10.1145/2766462.2767763>
- Schedl, M., Knees, P., McFee, B., Bogdanov, D., & Kaminskas, M.** (2015). Music recommender systems. In *Recommender Systems Handbook*, pages 453–492. Springer. DOI: https://doi.org/10.1007/978-1-4899-7637-6_13
- Schedl, M., Zamani, H., Chen, C.-W., Deldjoo, Y., & Elahi, M.** (2018b). Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, 7(2), 95–116. DOI: <https://doi.org/10.1007/s13735-018-0154-2>
- Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M.** (2002). Methods and metrics for coldstart recommendations. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260. ACM. DOI: <https://doi.org/10.1145/564376.564421>
- Schubert, E.** (2007). The influence of emotion, locus of emotion and familiarity upon preference in music. *Psychology of Music*, 35(3), 499–515. DOI: <https://doi.org/10.1177/0305735607072657>
- Seitlinger, P., Kowald, D., Kopeinik, S., Hasani-Mavriqi, I., Lex, E., & Ley, T.** (2015). Attention please! A hybrid resource recommender mimicking attention-interpretation dynamics. In *Companion Proceedings of International World Wide Web Conference*, pages 339–345. ACM. DOI: <https://doi.org/10.1145/2740908.2743057>
- Selvi, C., & Sivasankar, E.** (2019). An efficient context-aware music recommendation based on emotion and time context. In Mishra, D. K., Yang, X.-S., & Unal, A., Editors, *Data Science and Big Data Analytics*, pages 215–228. Springer. DOI: https://doi.org/10.1007/978-981-10-7641-1_18
- Shi, Y., Larson, M., & Hanjalic, A.** (2014). Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys*, 47(1), 3:1–3:45. DOI: <https://doi.org/10.1145/2556270>

- Vall, A., Quadrana, M., Schedl, M., & Widmer, G.** (2019). Order, context and popularity bias in next-song recommendations. *International Journal of Multimedia Information Retrieval*, 8(2), 101–113. DOI: <https://doi.org/10.1007/s13735-019-00169-8>
- van den Oord, A., Dieleman, S., & Schrauwen, B.** (2013). Deep content-based music recommendation. In *Proceedings of Neural Information Processing Systems Conference*, pages 2643–2651. Curran Associates Inc.
- Volkovs, M., Rai, H., Cheng, Z., Wu, G., Lu, Y., & Sanner, S.** (2018). Two-stage model for automatic playlist continuation at scale. In *Proceedings of ACM Conference on Recommender Systems*, page 9. ACM. DOI: <https://doi.org/10.1145/3267471.3267480>
- Zheng, E., Kondo, G. Y., Zilora, S., & Yu, Q.** (2018). Tag-aware dynamic music recommendation. *Expert Systems with Applications*, 106, 244–251. DOI: <https://doi.org/10.1016/j.eswa.2018.04.014>

How to cite this article: Lex, E., Kowald, D., & Schedl, M. (2020). Modeling Popularity and Temporal Drift of Music Genre Preferences. *Transactions of the International Society for Music Information Retrieval*, 3(1), pp. 17–30. DOI: <https://doi.org/10.5334/tismir.39>

Submitted: 19 June 2019

Accepted: 15 November 2019

Published: 25 March 2020

Copyright: © 2020 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.



Transactions of the International Society for Music Information Retrieval is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS

P3 Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations (2020)

Transparency and Cognitive Models in Recommender Systems

P3 Kowald, D.*, Lex, E.*, Schedl, M. (2020). Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations. In *4th Workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory (HUMANIZE @ ACM IUI'2020)*. (*equal contribution)

DOI: <https://doi.org/10.48550/arXiv.2003.10699>

Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations

Dominik Kowald*

Know-Center GmbH
Graz, Austria

dkowald@know-center.at

Elisabeth Lex*

Graz University of Technology
Graz, Austria

elisabeth.lex@tugraz.at

Markus Schedl

Johannes Kepler University Linz
Linz, Austria

markus.schedl@jku.at

ABSTRACT

In this paper, we introduce a psychology-inspired approach to model and predict the music genre preferences of different groups of users by utilizing human memory processes. These processes describe how humans access information units in their memory by considering the factors of (i) past usage frequency, (ii) past usage recency, and (iii) the current context. Using a publicly available dataset of more than a billion music listening records shared on the music streaming platform Last.fm, we find that our approach provides significantly better prediction accuracy results than various baseline algorithms for all evaluated user groups, i.e., (i) low-mainstream music listeners, (ii) medium-mainstream music listeners, and (iii) high-mainstream music listeners. Furthermore, our approach is based on a simple psychological model, which contributes to the transparency and explainability of the calculated predictions.

1 INTRODUCTION

Computational models of user preferences are crucial elements of music recommender systems [27] to tailor recommendations to the preferences of the user. Such user models are typically derived from the listening behavior of the users, i.e., their interactions with music artifacts, content features of music [34], or hybrid combinations of both. Research in music psychology [16] has shown that a wide range of factors impact music preferences [27], such as emotional state [5, 10], a user's current context [20], or a user's personality [20, 25]. Several aspects make the modeling of music preferences

challenging, such as, e.g., that music consumption is context-dependent and serves various purposes for listeners [28]. Also, recent research [7] has verified that classic music recommendation approaches suffer from popularity bias, i.e., they are biased to the mainstream that is prevalent in a music community. As a result, listeners of non-mainstream music receive less relevant recommendations compared to listeners of popular, mainstream music [4, 17, 22, 23].

In this paper, we introduce a psychology-inspired approach to model and predict the music genre preferences of users. We base our approach on research in music psychology that found music liking being positively influenced by prior exposure to the music [18, 29]. This has been attributed to the *mere exposure effect* or *familiarity principle* [33], i.e., users tend to establish positive preferences for items to which they are frequently and consistently exposed. Our idea is to computationally model prior exposure to music genres using the activation equation of human memory from the cognitive architecture *Adaptive Control of Thought–Rational* (ACT-R) [1, 2]. The activation equation determines the usefulness of a memory unit (i.e., its *activation*) for a user in the current context, based on how frequently and recently a user accessed it in the past as well as how important this unit is in the current context. In our previous work, we have employed a specific part of the activation equation, namely the Base-Level-Learning (BLL) equation, to recommend music artists [14]. The BLL equation computes the base-level activation of a memory unit based on how frequently and recently a user has accessed it in the past, following a time-dependent decay in the form of a power-law distribution. A high base-level activation means that the memory unit is vital for the user and, thus, can be more easily retrieved from her memory. However, in this work [14], we did not implement the full activation equation as we left out the associative activation that tunes the base-level activation of the memory unit to the current context.

In the present paper, we extend our previous model and utilize the associative activation for music genre predictions. This helps us tune the predictions to the current context of the user. As the current context, we utilize the set of genres

*Both authors contributed equally to this work.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

IUI '20 Workshops, March 17, 2020, Cagliari, Italy

© 2020 Copyright held by the owner/author(s).

<https://doi.org/https://doi.org/10.48550/arXiv.2003.10699>

User Group	$ U $	$ A $	$ G $	$ LE $	$ GA $	$ GA / LE $	$ G / U $	$Avg.MS$	$Avg.Age$	M/F
LowMS	1,000	82,417	931	6,915,352	14,573,028	2.107	85.771	.125	24.582	74%/26%
MedMS	1,000	86,249	933	7,900,726	20,264,870	2.565	126.439	.379	25.352	68%/32%
HighMS	1,000	92,690	973	8,251,022	22,498,370	2.727	186.010	.688	21.486	65%/35%

Table 1: Dataset statistics for the LowMS, MedMS, and HighMS Last.fm user groups. Here, $|U|$ is the number of distinct users, $|A|$ is the number of distinct artists, $|G|$ is the number of distinct genres, $|LE|$ is the number of listening events, $|GA|$ is the number of genre assignments, $|GA|/|LE|$ is the average number of genre assignments per LE, $|G|/|U|$ is the average number of genres a user has listened to, $Avg.MS$ is the average mainstreamness value, $Avg.Age$ is the average age of users in the group and M/F is the users' male/female ratio.

that are assigned to the most recently listened artist of a user. On a publicly available dataset of Last.fm music listening histories, we model the genre preferences of users from three different groups, which we extract using behavioral data in the form of music listening events: (i) LowMS, i.e., listeners of niche music (low mainstreamness), (ii) HighMS, i.e., listeners of mainstream music (high mainstreamness), and (iii) MedMS, i.e., listeners of music that lies in-between (medium mainstreamness). We introduce the $ACT_{u,a}$ approach that employs the full activation equation to take into account the current context of the user, which we define as the user's current genre preference. We compare the efficacy of $ACT_{u,a}$ to a variant, i.e., BLL_u , that uses only the BLL equation to model the past usage frequency (i.e., popularity) and recency (i.e., time). Furthermore, we compare both approaches to five baselines, including two collaborative filtering variants, mainstream-aware genre modeling, popularity-aware genre modeling, as well as time-based genre modeling.

The contributions of our work are two-fold. Firstly, we propose $ACT_{u,a}$, as an extension to BLL_u , to model and predict the genre preferences of users. Secondly, we evaluate the efficacy of both BLL_u and $ACT_{u,a}$ on three different groups of Last.fm users, which we separate based on the distance of their listening behavior to the mainstream: (i) LowMS, (ii) MedMS, and (iii) HighMS. We find that both BLL_u and $ACT_{u,a}$ outperform the five baseline methods in all three groups, with $ACT_{u,a}$ achieving the significantly highest performance. Our results also show that with both BLL_u and $ACT_{u,a}$, we can specifically improve the prediction performance for the users in the LowMS group. In other words, we can serve better the music consumers, whose prediction quality suffers the most from popularity bias. Also, both BLL_u and $ACT_{u,a}$ are based on a psychological theory, whose computational model is transparent and explainable and not a black box.

2 DATA AND APPROACH

In this section, we describe the Last.fm dataset as well as our music genre modeling and prediction approaches.

Dataset

In this paper, we use the publicly available *LFM-1b* dataset¹ of music listening information shared by users of the online music platform Last.fm. *LFM-1b* contains listening histories of more than 120,000 users, which sums up to over 1.1 billion listening events (LEs) collected between January 2005 and August 2014. Each LE contains a user identifier, the artist, the album, the track name, and a timestamp [21]. Furthermore, the *LFM-1b* dataset contains demographic data of the users such as country, age, gender, and a mainstreamness score, which is defined as the overlap between a user's personal listening history and the aggregated listening history of all Last.fm users in the dataset. Thus, the mainstreamness score reflects a user's inclination to music listened to by the Last.fm mainstream listeners (i.e., the "average" Last.fm listener) [26].

User groups. In order to study different types of users, we use this mainstreamness score to split the *LFM-1b* dataset into three equally sized user groups based on their mainstreamness (i.e., low, medium, and high). Specifically, we sort all users based on their mainstreamness score and assign the 1,000 users with the lowest scores to the low-mainstream group (i.e., *LowMS*), the 1,000 users with scores around the median mainstreamness (= .379) to the medium-mainstream group (i.e., *MedMS*), and the 1,000 users with the highest scores to the high-mainstream group (i.e., *HighMS*).

In our study, we consider only users with a minimum of 6,000 and a maximum of 12,000 LEs. We choose these thresholds based on the average number of LEs per user in the dataset, which is 9,043, as well as the kernel density distribution of the data. With this method, on the one hand, we exclude users with too little data available for training our algorithms (i.e., users with less than 6,000 LEs), and on the other hand, we exclude so-called power listeners (i.e., users with more than 12,000 LEs) that might distort our results. Table 1 summarizes the statistics and characteristics of our three user groups. We see that, even if we only consider 1,000 users per group, we have a sufficient amount of LEs, i.e., between 6.9 to 8.3 million, to train and test our music genre

¹<http://www.cp.jku.at/datasets/LFM-1b>

modeling and prediction approaches. Further characteristics of our user groups are as follows:

(i) **LowMS.** The LowMS group represents the $|U| = 1,000$ users with the smallest mainstreaminess scores. These users have an average mainstreaminess value of $\text{Avg.} MS = .125$. LowMS contains $|A| = 82,417$ distinct artists, $|LE| = 6,915,352$ listening events, $|G| = 931$ genres, and $|GA| = 14,573,028$ genre assignments. Interestingly, the male/female ratio is the least evenly distributed one in this group (i.e., $M/F = 74\%/26\%$).

(ii) **MedMS.** The MedMS group consists of the $|U| = 1,000$ users with mainstreaminess scores around the median and thus, lying between the ones of the LowMS and HighMS groups. This group has an average mainstreaminess value of $\text{Avg.} MS = .379$. The majority of dataset statistics of this group lies between the ones of the LowMS and HighMS users, except for the average age, which is the highest for the MedMS users (i.e., $\text{Avg.} Age = 25.352$ years).

(iii) **HighMS.** The HighMS group represents the $|U| = 1,000$ users in the *LFM-1b* dataset with the highest mainstreaminess scores ($\text{Avg.} MS = .688$). These users are not only the youngest ones (i.e., $\text{Avg.} Age = 21.486$ years) but also listen to the highest number of distinct genres on average (i.e., $|G|/|U| = 186.010$), indicating that music which is considered mainstream is quite diverse on Last.fm. Also, this user group exhibits the largest number of female listeners (i.e., $M/F = 65\%/35\%$) and the highest number of distinct genres ($|G| = 973$).

Additionally, we investigate the most frequent countries of the users. Here, for all three groups, the United States (US) is the dominating country. The share of US users increases with the mainstreaminess, i.e., while this share is only 14% for LowMS and 18% for MedMS, it is already 22% for HighMS. Interestingly, Russia (RU, 13%), Poland (PL, 9%), and Japan (JP, 8%) are frequent in the LowMS group, while the United Kingdom (UK) contributes a substantial share in the other two groups (9% for MedMS and 14% for HighMS). Germany (DE) is among the most popular countries in all three groups (10% for LowMS and HighMS, 8% for MedMS); Brazil (BR) can only be found among the most popular countries in the MedMS group (8%); and the Netherlands (NL, 5%) as well as Spain (ES, 4%) can only be found in the HighMS group.

Genre mapping. For mapping music genres to artists, we use an extension of the *LFM-1b* dataset, namely the *LFM-1b UGP* dataset [24], which describes the genres of an artist by leveraging social tags assigned by Last.fm users. Specifically, *LFM-1b UGP* contains a weighted mapping of 1,998 music genres available in the online database Freebase² to Last.fm artists. This database includes a fine-grained representation of musical styles, including genres such as “Progressive Psy-trance” or “Pagan Black Metal”.

²<https://developers.google.com/freebase/> (no longer maintained)

The genre weightings for any given artist correspond to the relative frequency of tags assigned to that artist in Last.fm. For example, for the artist “Metallica”, the top tags and their corresponding relative frequencies are “thrash metal” (1.0), “metal” (.91), “heavy metal” (.74), “hard rock” (.41), “rock” (.34), and “seen live” (.3). From this list, we remove all tags that are not part of the 1,998 Freebase genres (i.e., “seen live” in our example) as well as all tags with a relative frequency smaller than .5 (i.e., “hard rock” and “rock” in our example). Thus, for “Metallica”, we end up with three genres, i.e., “thrash metal”, “metal” and “heavy metal”.

Approach

In this section, we describe our music genre modeling and prediction approach based on the declarative memory module of ACT-R.

The Cognitive Architecture ACT-R. ACT-R, which is short for “Adaptive Control of Thought – Rational”, is a cognitive architecture developed by John Robert Anderson [1]. ACT-R defines and formalizes the basic cognitive operations of the human mind (e.g., access to information in human memory).

Figure 1 schematically illustrates the main architecture of ACT-R. In general, ACT-R differs between short-term memory modules, such as the working memory module, and long-term memory modules, such as the declarative and procedural memory modules. Using a sensory register (i.e., the ultra-short-term memory), the encoded information is passed to the short-term working memory module, which interacts with the long-term memory modules. In the case of the declarative memory, the encoded information can be stored, and already stored information can be retrieved. In the case of the procedural memory, the information can be matched against stored rules that can lead to actions [32].

Thus, declarative memory holds factual knowledge (e.g., what something is), and procedural memory consists of sequences of actions (e.g., how to do something). In our work, we focus on the declarative part, which contains the activation equation of human memory. The activation equation determines the usefulness, i.e., the activation level A_i , of a memory unit i (e.g., a music genre in our case) for a user u in the current context. It is given by:

$$A_i = B_i + \sum_j W_j \cdot S_{j,i} \quad (1)$$

Here, the B_i component represents the *base-level* activation and quantifies the general usefulness of the unit i by considering how frequently and recently it has been used in the past. It is given by the base-level learning (BLL) equation:

$$B_i = \ln \left(\sum_{j=1}^n t_j^{-d} \right) \quad (2)$$

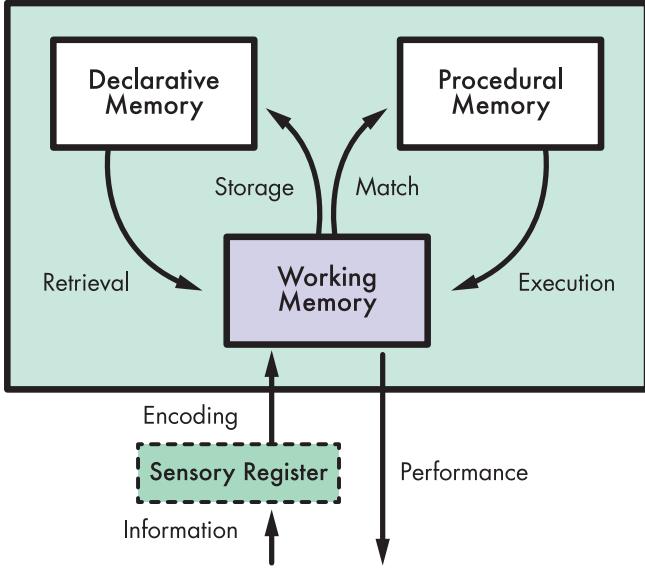


Figure 1: Schematic illustration of ACT-R. In our work, we focus on the activation equation of the declarative memory module.

where n is the frequency of i 's occurrences and t_j is the time since the j^{th} occurrence of i . The exponent d accounts for the power-law of forgetting, which means that each unit's activation level caused by the j^{th} occurrence decreases in time according to a power function [1].

The second component of Equation 1 represents the *associative activation* that tunes the base-level activation of the unit i to the current context. The context is given by any contextual element j that is relevant for the current situation. In the case of a music recommender system, that could be a music genre that the user prefers in the current situation. Through learned associations, the contextual elements are connected with i and can increase i 's activation depending on the weight W_j and the strength of association $S_{j,i}$.

Modeling and Predicting Music Genre Preferences. For modeling and predicting music genre preferences, we investigate two approaches: (i) BLL_u based on the BLL equation to model the past usage frequency (i.e., popularity) and recency (i.e., time), and (ii) $ACT_{u,a}$ based on the full activation equation to also take the current context into account.

We start with BLL_u and thus, with defining the base-level activation $B(g, u)$ for genre g and user u by utilizing the previously defined BLL equation:

$$B(g, u) = \ln \left(\sum_{j=1}^n t_{u,g,j}^{-d} \right) \quad (3)$$

Here, g is a genre user u has listened to in the past, and n is the number of times u has listened to g . Further, $t_{u,g,j}$ is

the time in seconds since the j^{th} LE of g by u , and d is the power-law decay factor, which we identify using a similar method as used in [15]. Thus, in Figure 2, for all LEs and genres in our dataset, we plot the relistening count of a genre g over the time since the last LE of g . Then, we set d to the slope α of the linear regression lines of this data, which leads to 1.480 for LowMS, 1.574 for MedMS, and 1.587 for HighMS.

The resulting base-level activation values $B(g, u)$ are then normalized using a simple softmax function in order to map them onto a range of $[0, 1]$ that sums up to 1 [13, 15]:

$$B'(g, u) = \frac{\exp(B(g, u))}{\sum_{g' \in G_u} \exp(B(g', u))} \quad (4)$$

Here, G_u is the set of distinct genres listened to by u . Finally, BLL_u predicts the top- k genres \widetilde{G}_u^k with highest $B'(g, u)$ values to u :

$$\widetilde{G}_u^k = \underbrace{\arg \max_{g \in G_u} (B'(u, g))}_{BLL_u} \quad (5)$$

To investigate not only the factors of frequency and time but also the current context by means of an associative activation, we implement the full activation equation (see Equation 1) in the form of:

$$A(g, u, a) = B'(g, u) + \sum_{c \in G_a} W_c \cdot S_{c,g} \quad (6)$$

where the first part represents the base-level activation by means of the BLL equation and the second part represents the associative activation.

To calculate the associative activation and thus, to model a user's current context, we incorporate the set of genres G_a assigned to the most recently listened to artist a by user u . When applying this equation in the context of recommender systems, related literature [31] suggests using a measure of normalized co-occurrence to represent the strength of an association $S_{c,g}$. Accordingly, we define the co-occurrence between two genres as the number of artists to which both genres are assigned. We normalize this co-occurrence value according to the Jaccard coefficient:

$$S_{c,g} = \frac{|A_c \cap A_g|}{|A_c \cup A_g|} \quad (7)$$

where A_c is the set of artists to which context-genre c is assigned, and A_g is the set of artists to which genre g is assigned. Thus, we set the number of times two genres co-occur into relation with the number of times in which at least one of the two genres appears. In this work, we set the attentional weight W_c of context-genre c to 1. By doing so, we give equal weights to all genres assigned to an artist, which avoids down-ranking of less popular, but perhaps more specific, and hence more valuable, genres.

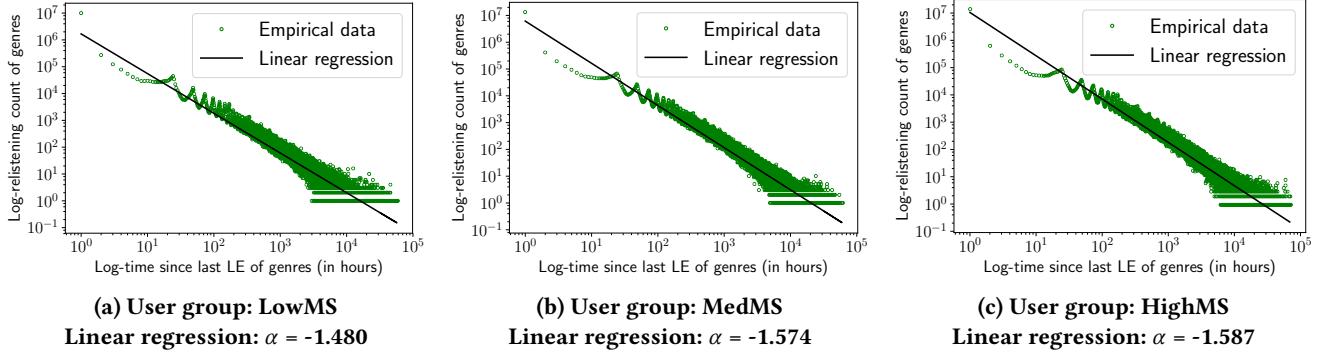


Figure 2: Calculation of the BLL equation’s d parameter. On a log-log scale, we plot the relistening count of the genres over the time since their last LEs. We set d to the slopes α of the corresponding linear regression lines.

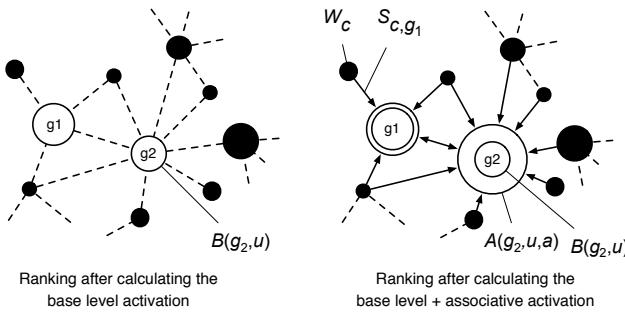


Figure 3: Example illustrating the difference between BLL_u (left panel) and $ACT_{u,a}$ (right panel). Here, unfilled nodes represent target genres g_1 and g_2 , and black nodes represent genres of the last artist listened to by the target user (i.e., contextual genres). For g_1 and g_2 , the node sizes represent the activation levels and for the contextual genres, the node sizes represent the attentional weights W_c . The association strength $S_{c,g}$ is represented by the edge lengths. While BLL_u determines a higher activation level for g_1 than for g_2 , $ACT_{u,a}$ gives a higher activation level to g_2 than to g_1 by also considering the associative association based on the current context.

Finally, we normalize the $A(g, u, a)$ values using the aforementioned softmax function and predict the top- k genres \widetilde{G}_u^k with highest $A'(g, u, a)$ values for a given user u and the genres of the user’s most recently listened artist a (i.e., the current context):

$$\widetilde{G}_u^k = \underbrace{\arg \max_{g \in G_u} (A'(u, g, a))}_{ACT_{u,a}} \quad (8)$$

We further illustrate the difference between BLL_u and $ACT_{u,a}$ in the example of Figure 3 by showing the additional

impact of the associative activation defined by the second component of the activation equation. As defined, this associative activation is evoked by the current context (i.e., the genres of the last artist the target user has listened to).

The left panel of Figure 3 shows two genres, g_1 and g_2 , with different base-level activation levels (illustrated by the circle size). Thus, according to BLL_u , g_1 reaches a higher base-level activation, which means a better rank, than g_2 . This relationship changes in the right panel of Figure 3, where we consider the influence of the genres in the current context (illustrated by the black nodes). Specifically, depending on the weights W_c (represented by the size of the black nodes) and strength of association $S_{c,g}$ (represented by the length of the edges), the genres in the current context spread additional associative activation to the genres g_1 and g_2 . Now, according to $ACT_{u,a}$, g_2 receives stronger associative activation than g_1 , which also leads to a better rank.

3 EXPERIMENTS AND RESULTS

In this section, we describe our experimental setup, i.e., the baseline algorithms, the evaluation protocol and metrics, as well as the results of our experiments.

Baseline Algorithms

We compare the BLL_u and $ACT_{u,a}$ approaches to five baseline algorithms:

Mainstream-based baseline: TOP. The *TOP* approach models a user u ’s music genre preferences using the overall top- k genres of all users (i.e., the mainstream) in u ’s user group (i.e., LowMS, MedMS, HighMS). This is given by:

$$\widetilde{G}_u^k = \arg \max_{g \in G} (|GA_g|) \quad (9)$$

Here \widetilde{G}_u^k denotes the set of k predicted genres, G the set of all genres, and $|GA_g|$ corresponds to the number of times g occurs in all genre assignments GA of u 's user group.

User-based collaborative filtering baseline: CF_u . User-based collaborative filtering-based approaches aim to find similar users for target user u (i.e., the set of neighbors N_u) and predict the genres these similar users have listened to in the past [30]. CF_u is given by:

$$\widetilde{G}_u^k = \arg \max_{g \in G(N_u)} \left(\sum_{v \in N_u} sim(G_u, G_v) \cdot |GA_{g,v}| \right) \quad (10)$$

where \widetilde{G}_u^k denotes the set of k predicted genres for user u , $G(N_u)$ are the genres listened to by the set of neighbors N_u ,³ $sim(G_u, G_v)$ is the cosine similarity between the genre distributions of user u and neighbor v . Finally, $|GA_{g,v}|$ indicates how often v has listened to genre g in the past.

Item-based collaborative filtering baseline: CF_i . Similar to CF_u , CF_i is a collaborative filtering-based approach, but instead of finding similar users for the target user u , it aims to find similar items, i.e., music artists S_{A_u} , for the artists A_u that u has listened to in the past. Then, it predicts the genres that are assigned to these similar artists as given by:

$$\widetilde{G}_u^k = \arg \max_{g \in G(S_{A_u})} \left(\sum_{a \in A_u} \sum_{s \in S_a} sim(G_a, G_s) \right) \quad (11)$$

where $G(S_{A_u})$ are the genres assigned to the similar artists S_{A_u} , S_a is the set of similar artists for an artist $a \in A_u$,⁴ and $sim(G_a, G_s)$ is the cosine similarity between the genre distributions assigned to a and the genres assigned to a similar artist $s \in S_a$.

Popularity-based baseline: POP_u . POP_u is a personalized music genre modeling technique, which predicts the k most frequently listened genres in the listening history of user u . POP_u is given by the following equation:

$$\widetilde{G}_u^k = \arg \max_{g \in G_u} (|GA_{g,u}|) \quad (12)$$

Here, G_u is the set of genres u has listened to in the past and $|GA_{g,u}|$ denotes the number of times u has listened to g . Thus, it ranks the genres u has listened to in the past by popularity.

Time-based baseline: $TIME_u$. The time-based baseline $TIME_u$ predicts the k genres that user u has most recently listened

³We set the neighborhood size for CF_u and CF_i to 20.

⁴For A_u , we consider the set of the 20 artists that u has listened to most frequently.

to. It is given by:

$$\widetilde{G}_u^k = \arg \min_{g \in G_u} (t_{u,g,n}) \quad (13)$$

where $t_{u,g,n}$ is the time since the last (i.e., the n^{th}) LE of g by u .

Evaluation Protocol and Metrics

We split the datasets into train and test sets [6]. In doing so, we ensure that our evaluation protocol preserves the temporal order of the LEs, which simulates a real-world scenario in which we predict genres of future LEs based on past ones and not the other way round [15]. This also means that a classic k -fold cross-validation evaluation protocol is not useful in our setting.

Specifically, we put the most recent 1% of the LEs of each user into the test set (i.e., LE_{test}) and keep the remaining LEs for the train set (i.e., LE_{train}). We do not use a classic 80/20 split as the number of LEs per user is large (i.e., on average, 7,689 LEs per user). Although we only use the most recent 1% of listening events per user, this process leads to three large test sets with 69,153 listening events for LowMS, 79,007 listening events for MedMS, and 82,510 listening events for HighMS. To finally measure the prediction quality of the approaches, we use the following six well-established performance metrics [3]:

Recall: $R@k$. Recall is calculated as the number of correctly predicted genres divided by the number of relevant genres taken from the LEs in the test set LE_{test} . It is a measure for the completeness of the predictions and is formally given by:

$$R@k = \frac{1}{|LE_{test}|} \sum_{u,a \in LE_{test}} \frac{|\widetilde{G}_u^k \cap G_{u,a}|}{|G_{u,a}|} \quad (14)$$

where \widetilde{G}_u^k denotes the k predicted genres and $G_{u,a}$ the set of relevant genres of an artist a in user u 's LEs in the test set.

Precision: $P@k$. Precision is calculated as the number of correctly predicted genres divided by the number of predictions k and is a measure for the accuracy of the predictions. It is given by:

$$P@k = \frac{1}{|LE_{test}|} \sum_{u,a \in LE_{test}} \frac{|\widetilde{G}_u^k \cap G_{u,a}|}{k} \quad (15)$$

We report recall and precision for $k = 1 \dots 10$ predicted genres in form of recall/precision plots.

F1-score: $F1@k$. F1-score is the harmonic mean of recall and precision:

$$F1@k = 2 \cdot \frac{P@k \cdot R@k}{P@k + R@k} \quad (16)$$

User group	Evaluation metric	<i>TOP</i>	<i>CF_u</i>	<i>CF_i</i>	<i>POP_u</i>	<i>TIME_u</i>	<i>BLL_u</i>	<i>ACT_{u,a}</i>
LowMS	<i>F1@5</i>	.108	.311	.341	.356	.368	.397	.485***
	<i>MRR@10</i>	.101	.389	.425	.443	.445	.492	.626***
	<i>MAP@10</i>	.112	.461	.505	.533	.550	.601	.785***
	<i>nDCG@10</i>	.180	.541	.590	.618	.625	.679	.824***
MedMS	<i>F1@5</i>	.196	.271	.284	.292	.293	.338	.502***
	<i>MRR@10</i>	.146	.248	.264	.274	.272	.320	.511***
	<i>MAP@10</i>	.187	.319	.336	.351	.365	.419	.705***
	<i>nDCG@10</i>	.277	.419	.441	.460	.452	.523	.753***
HighMS	<i>F1@5</i>	.247	.273	.266	.282	.228	.304	.427***
	<i>MRR@10</i>	.188	.232	.229	.242	.201	.266	.412***
	<i>MAP@10</i>	.246	.304	.298	.314	.267	.348	.569***
	<i>nDCG@10</i>	.354	.413	.402	.429	.357	.462	.642***

Table 2: Genre prediction accuracy results comparing our BLL_u and $ACT_{u,a}$ approaches with a mainstream-based baseline (TOP), a user-based collaborative filtering baseline (CF_u), an item-based collaborative filtering baseline (CF_i), a popularity-based baseline (POP_u) and a time-based baseline ($TIME_u$). For all three user groups (i.e., LowMS, MedMS, and HighMS), $ACT_{u,a}$ outperforms all other approaches. According to a t-test with $\alpha = .001$, “*” indicates statistically significant differences between $ACT_{u,a}$ and all other approaches.**

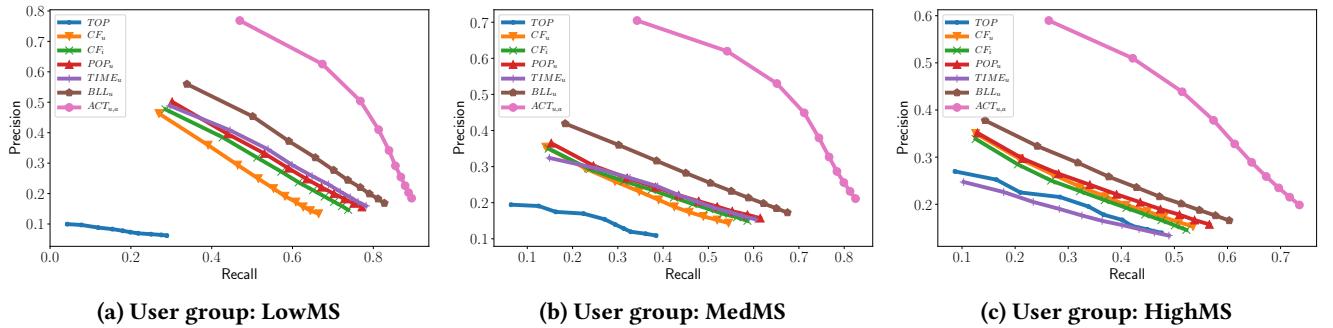


Figure 4: Recall/precision plots for $k = 1 \dots 10$ predicted genres of the baselines and our BLL_u and $ACT_{u,a}$ approaches for the three user groups LowMS, MedMS, and HighMS. $ACT_{u,a}$ achieves the best results in all settings.

We report the F1-score for $k = 5$, where it typically reaches its highest value if 10 genres are predicted.

Mean Reciprocal Rank: MRR@k. MRR is the average of reciprocal ranks $r(g)$ of all relevant genres in the list of predicted genres:

$$MRR@k = \frac{1}{|LE_{test}|} \sum_{u,a \in LE_{test}} \frac{1}{|G_{u,a}|} \sum_{g \in G_{u,a}} \frac{1}{r(g)} \quad (17)$$

This means that a high MRR is achieved if relevant genres occur at the beginning of the predicted genre list.

Mean Average Precision: MAP@k. MAP is an extension of the precision metric by also taking the ranking of the

correctly predicted genres into account and is given by:

$$MAP@k = \frac{1}{|LE_{test}|} \sum_{u,a \in LE_{test}} \frac{1}{|G_{u,a}|} \sum_{i=1}^k Rel_i \cdot P@i \quad (18)$$

Here, Rel_i is 1 if the predicted genre at position i is among the relevant genres (0 otherwise) and $P@i$ is the precision calculated at position i according to Equation 15.

Normalized Discounted Cumulative Gain: nDCG@k. nDCG is another ranking-dependent metric. It is based on the Discounted Cumulative Gain ($DCG@k$) measure [9], which is defined as:

$$DCG@k = \sum_{i=1}^k \left(\frac{2^{Rel_i} - 1}{\log_2(1 + i)} \right) \quad (19)$$

where Rel_i is 1 if the genre predicted for the i^{th} item is relevant (0 otherwise). $nDCG@k$ is given as $DCG@k$ divided by $iDCG@k$, which is the highest possible DCG value that can be achieved if all relevant genres are predicted in the correct order:

$$nDCG@k = \frac{1}{|LE_{test}|} \sum_{u,a \in LE_{test}} \left(\frac{DCG@k}{iDCG@k} \right) \quad (20)$$

We report MRR, MAP, and nDCG for $k = 10$ predicted music genres, where these metrics reach their highest values.

Results and Discussion

In this section, we present and discuss our evaluation results. The accuracy results according to $F1@5$, $MRR@10$, $MAP@10$, and $nDCG@10$ are shown in Table 2 for the five baseline approaches as well as the proposed BLL_u and $ACT_{u,a}$ algorithms. Furthermore, we provide recall/precision plots for $k = 1 \dots 10$ predicted genres.

Accuracy of baseline approaches. When analyzing the performance of the baseline approaches TOP , CF_u , CF_i , POP_u , and $TIME_u$, we see a clear difference between the non personalized and the personalized algorithms. While the non personalized TOP approach, which predicts the top- k genres of the mainstream, provides better accuracy results in the HighMS setting than in the LowMS setting, the personalized CF_u , CF_i , POP_u , and $TIME_u$ algorithms provide better results in the LowMS setting than in the HighMS setting. Hence, personalized genre modeling approaches provide better results, the lower the mainstreaminess of the users. Non-personalized genre modeling approaches, however, have higher performance, the higher the mainstreaminess of the users.

Next, we compare the accuracy of the two collaborative filtering-based methods, CF_u , and CF_i . Here, the item-based CF variant CF_i reaches higher accuracy estimates in the LowMS and MedMS settings, while the user-based CF variant CF_u provides better performance in the HighMS setting. To better understand this pattern of results, we provide the average pairwise user similarity in the form of boxplots in Figure 5. Here, for all three user groups, we calculate the pairwise similarity between the users via the cosine similarity metric based on the users' genre distribution vectors. We see that users in the HighMS setting are very similar to each other, which explains the good performance of an algorithm that is based on user similarities, such as CF_u .

POP_u and $TIME_u$ reach the highest accuracy estimates among the five baseline approaches. Interestingly, the popularity-based POP_u algorithm provides the best results for the HighMS user group, while the time-based $TIME_u$ algorithm provides the best results in the LowMS user group. For the MedMS user group, however, both algorithms reach a comparable

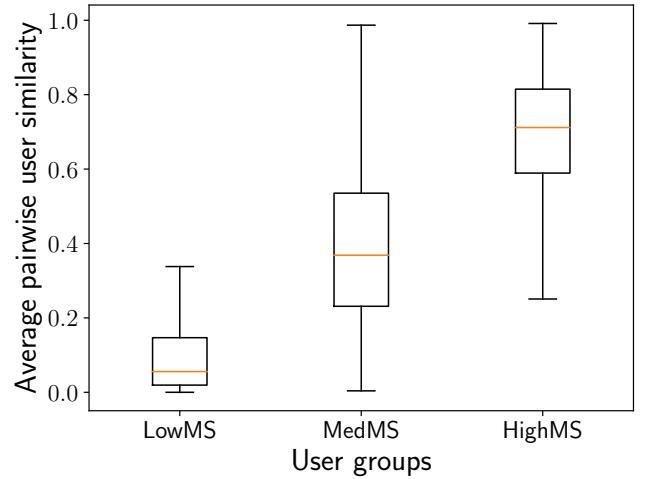


Figure 5: Average pairwise user similarity for LowMS, MedMS, and HighMS. We calculate the user similarity using the cosine similarity metric based on the users' genre distributions. While users in the LowMS group show a very individual listening behavior, users in the HighMS group tend to listen to similar music genres.

accuracy performance, which shows the importance of both factors, frequency (i.e., popularity) and recency (i.e., time).

Accuracy of BLL_u and $ACT_{u,a}$. We discuss the results of the BLL_u and $ACT_{u,a}$ approaches, which utilize human memory processes as defined by the cognitive architecture ACT-R in order to model and predict music genre preferences. Specifically, BLL_u combines the factors of past usage frequency and recency via the BLL equation (see Equation 3) and $ACT_{u,a}$ extends BLL_u by also considering the current context via the activation equation (see Equation 6). In this work, we define the current context by the genres assigned to the artist that the target user u has listened to most recently.

As expected, when combining the factors of past usage frequency and recency in the form of BLL_u , we can outperform the best performing baseline approaches POP_u and $TIME_u$ in all three settings (i.e., LowMS, MedMS, and HighMS). We can further improve the accuracy performance when we additionally consider the current context in the form of $ACT_{u,a}$. Here, we reach a statistically significant improvement⁵ over all other approaches across all evaluation metrics and user groups. Furthermore, in Figure 6, we present a recall/precision plot showing the accuracy of $ACT_{u,a}$ for $k = 1 \dots 10$ predicted genres for LowMS, MedMS, and HighMS. We observe good results for all three user groups but especially in the LowMS setting, in which we are faced with users with a low interest in mainstream music.

⁵According to a t-test with $\alpha = .001$.

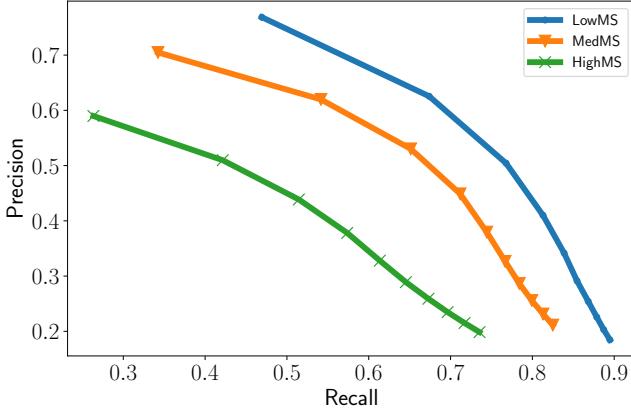


Figure 6: Recall/precision plot of our $ACT_{u,a}$ approach for $k = 1 \dots 10$ predicted genres for the three user groups LowMS, MedMS, and HighMS. We observe good prediction accuracy results for $ACT_{u,a}$ in all settings but especially for LowMS. This shows that our approach based on human memory processes is especially useful for predicting the music genre preferences of users with low interest in mainstream music.

This shows that the proposed $ACT_{u,a}$ algorithm can provide accurate predictions of music genres listened to in the future for all user groups and, thus, treats all users in our experiment in a fair manner. Moreover, since our approach utilizes human memory processes, it is based on psychological principles of human intelligence rather than artificial intelligence. We believe that this theoretical underpinning contributes to the explanation effectiveness of our approach as we can fully understand why a specific genre was predicted for a target user in a given context. To further illustrate this with an example, we would like to refer back to Figure 3. In this figure, we have shown the differences between BLL_u and $ACT_{u,a}$ for two predicted genres g_1 and g_2 . Let us assume that these are the top-2 predicted genres for a target user u . According to BLL_u , we know that these genres got the highest activation levels because u has listened to them very frequently and recently. When looking at the activation levels calculated by $ACT_{u,a}$, we also take the current context into account and, thus, get an indication for the similarity of g_1 and g_2 to the genres assigned to the most recently listened artist a of user u . In our example, genre g_2 is strongly related to the current context, while genre g_1 only has a weak relation to it. Taken together, with our $ACT_{u,a}$ approach, we can easily explain genre prediction results according to three simple factors that are relevant for human memory processes according to the cognitive architecture ACT-R: (i) past usage frequency, (ii) past usage recency, and (iii) similarity to current context.

4 CONCLUSION AND FUTURE WORK

In this paper, we presented BLL_u and $ACT_{u,a}$, two music genre preference modeling, and prediction approaches based on the human memory module of the cognitive architecture ACT-R. While BLL_u utilizes the BLL equation of ACT-R in order to model the factors of past usage frequency (i.e., popularity) and recency (i.e., time), $ACT_{u,a}$ integrates the activation equation of ACT-R to also incorporate the current context. We defined this context as the genres assigned to the most recently listened artist of the target user.

Using a dataset gathered from the music platform Last.fm, we evaluated BLL_u and $ACT_{u,a}$ against a mainstream-based approach TOP , a user-based CF approach CF_u , an item-based CF approach CF_i , a popularity-based approach POP_u as well as a time-based approach $TIME_u$. We used six evaluation metrics (i.e., recall, precision, F1-score, MRR, MAP, and nDCG) in three evaluation settings in which the evaluated users differed in terms of their inclination to mainstream music (i.e., LowMS, MedMS, and HighMS user groups). Our evaluation results show that both BLL_u and $ACT_{u,a}$ outperform the five baseline methods in all three settings; $ACT_{u,a}$ even does so in a statistically significant manner. Furthermore, we find that especially the current context is of high importance when aiming for accurate genre predictions.

Summed up, in this work, we have shown that human memory processes in the form of ACT-R's activation equation can be effectively utilized for modeling and predicting music genres. By following such a psychology-inspired approach, we also believe that we can model a user's preferences transparently, in contrast to, e.g., deep learning-based approaches based on latent user representations. Therefore, our approach could be useful to realize more transparent and explainable music recommender systems.

Limitations and future work. In the present work, we only considered the genres assigned to the most recently listened artist of the target user as contextual information. However, related work on music preference modeling has shown that music listening habits depend on the time of the day, the current activity of a user or the mood a user is currently experiencing (see, e.g., [11]).

For future work, we also plan to utilize the procedural memory processes of ACT-R in addition to the activation equation. As, for instance, done in the SNIF-ACT model [8, 19], we could define so-called production rules in order to transfer the user's preferences into actual music recommendation strategies. By making these rules transparent to the user, we aim to contribute to research on transparent recommender systems that create explainable recommendations.

Reproducibility. To foster the reproducibility of our research, we use the publicly available LFM-1b dataset (see

Section 2). Furthermore, we provide the source code of our approach as part of our TagRec framework [12].

REFERENCES

- [1] John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. 2004. An integrated theory of the mind. *Psychological review* 111, 4 (2004), 25 pages.
- [2] John R Anderson and Lael J Schooler. 1991. Reflections of the environment in memory. *Psychological science* 2, 6 (1991), 396–408.
- [3] Ricardo Baeza-Yates, Berthier de Araújo Neto Ribeiro, et al. 2011. *Modern Information Retrieval*. New York: ACM Press; Harlow, England: Addison-Wesley.
- [4] Christine Bauer and Markus Schedl. 2019. Global and country-specific mainstreaminess measures: Definitions, analysis, and usage for improving personalized music recommendation systems. *PloS one* 14, 6 (2019), e0217389.
- [5] Joanne R Cantor and Dolf Zillmann. 1973. The effect of affective state and emotional arousal on music appreciation. *The Journal of General Psychology* 89, 1 (1973), 97–108.
- [6] Paolo Cremonesi, Roberto Turrin, Eugenio Lentini, and Matteo Matteucci. 2008. An Evaluation Methodology for Collaborative Recommender Systems. In *Proceedings of AXMEDIS'2008*. IEEE Computer Society, Washington, DC, USA, 224–231. <https://doi.org/10.1109/AXMEDIS.2008.13>
- [7] Kowald Dominik, Schedl Markus, and Lex Elisabeth. 2020. The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In *Proceedings of the 42nd European Conference on Information Retrieval*.
- [8] Wai-Tat Fu and Peter Pirolli. 2007. SNIF-ACT: A cognitive model of user navigation on the World Wide Web. *Human–Computer Interaction* 22, 4 (2007), 355–412.
- [9] Kalervo Järvelin, Susan L Price, Lois ML Delcambre, and Marianne Lykke Nielsen. 2008. Discounted cumulated gain based evaluation of multiple-query IR sessions. In *Proceedings of ECIR'2008*. Springer, 4–15.
- [10] Patrik N Juslin and John A Sloboda. 2001. *Music and emotion: Theory and research*. Oxford University Press.
- [11] P. Knees, M. Schedl, B. Ferwerda, and A. Laplante. 2019. User Awareness in Music Recommender Systems. In *Mirjam Augstein, Eelco Herder, Wolfgang Wörndl (eds.), Personalized Human-Computer Interaction*. De Gruyter.
- [12] Dominik Kowald, Simone Kopeinik, and Elisabeth Lex. 2017. The tagrec framework as a toolkit for the development of tag-based recommender systems. In *Adjunct Publication of UMAP'2017*. ACM, 23–28.
- [13] Dominik Kowald and Elisabeth Lex. 2016. The Influence of Frequency, Recency and Semantic Context on the Reuse of Tags in Social Tagging Systems. In *Proceedings of Hypertext'2016*. ACM, New York, NY, USA, 237–242.
- [14] Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2019. Modeling Artist Preferences for Personalized Music Recommendations. In *Proc. of ISMR '19*.
- [15] Dominik Kowald, Subhash Chandra Pujari, and Elisabeth Lex. 2017. Temporal Effects on Hashtag Reuse in Twitter: A Cognitive-Inspired Hashtag Recommendation Approach. In *Proceedings of WWW'2017*. ACM, 10 pages.
- [16] Adrian North and David Hargreaves. 2008. *The social and applied psychology of music*. OUP Oxford.
- [17] Aäron van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep Content-based Music Recommendation. In *Proceedings of NIPS'2013*. Curran Associates Inc., USA, 2643–2651.
- [18] Carlos Silva Pereira, João Teixeira, Patrícia Figueiredo, João Xavier, São Luís Castro, and Elvira Brattico. 2011. Music and emotions in the brain: familiarity matters. *PloS one* 6, 11 (2011), e27241.
- [19] Peter Pirolli and Wai-Tat Fu. 2003. SNIF-ACT: A model of information foraging on the World Wide Web. In *International Conference on User Modeling*. Springer, 45–54.
- [20] Peter J Rentfrow and Samuel D Gosling. 2003. The do re mi's of everyday life: the structure and personality correlates of music preferences. *Journal of personality and social psychology* 84, 6 (2003), 21 pages.
- [21] Markus Schedl. 2016. The LFM-1b Dataset for Music Retrieval and Recommendation. In *Proceedings of the 2016 Conference on Multimedia Retrieval*. ACM, 103–110.
- [22] Markus Schedl and Christine Bauer. 2017. Distance-and Rank-based Music Mainstreaminess Measurement. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*. ACM, 364–367.
- [23] Markus Schedl and Christine Bauer. 2018. An Analysis of Global and Regional Mainstreaminess for Personalized Music Recommender Systems. *Journal of Mobile Multimedia* 14 (2018), 95–112.
- [24] Markus Schedl and Bruce Ferwerda. 2017. Large-scale Analysis of Group-specific Music Genre Taste From Collaborative Tags. In *Proceedings of ISM'2017*. IEEE, 479–482.
- [25] Markus Schedl, Emilia Gómez, Erika Trent, Marko Tkalcic, Hamid Eghbal-Zadeh, and Agustín Martorell. 2018. On the Interrelation between Listener Characteristics and the Perception of Emotions in Classical Orchestra Music. *IEEE Transactions on Affective Computing* 9 (2018), 507–525. Issue 4.
- [26] Markus Schedl and David Hauger. 2015. Tailoring music recommendations to users by considering diversity, mainstreaminess, and novelty. In *Proceedings of SIGIR'2015*. ACM, 947–950.
- [27] Markus Schedl, Peter Knees, Brian McFee, Dmitry Bogdanov, and Marius Kamienskas. 2015. Music recommender systems. In *Recommender systems handbook*. Springer, 453–492.
- [28] Markus Schedl, Hamed Zamani, Ching-Wei Chen, Yashar Deldjoo, and Mehdi Elahi. 2018. Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval* 7, 2 (01 Jun 2018), 95–116.
- [29] Emery Schubert. 2007. The influence of emotion, locus of emotion and familiarity upon preference in music. *Psychology of Music* 35, 3 (2007), 499–515.
- [30] Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Collaborative Filtering Beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges. *Comput. Surveys* 47, 1, Article 3 (May 2014), 45 pages.
- [31] Leendert Van Maanen and Julian N Marewski. 2009. Recommender systems for literature selection: A competition between decision making and memory models. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*. 2914–2919.
- [32] Steve Wheeler. 2014. Learning Theories: Adaptive Control of Thought. [Online under <http://www.teachthought.com/learning/theory-cognitive-architecture/>; accessed 19-December-2019].
- [33] Robert B Zajonc. 1968. Attitudinal effects of mere exposure. *Journal of personality and social psychology* 9, 2p2 (1968), 1.
- [34] Eva Zangerle and Martin Pichl. 2018. Content-based User Models: Modeling the Many Faces of Musical Preference. In *19th International Society for Music Information Retrieval Conference*.

P4 Psychology-informed Recommender Systems (2021)
Transparency and Cognitive Models in Recommender Systems

[P4] Lex, E., **Kowald, D.**, Seitlinger, P., Tran, T., Felfernig, A., Schedl, M. (2021). Psychology-informed Recommender Systems. *Foundations and Trends in Information Retrieval*, 15:2, pp. 134–242.
DOI: <https://doi.org/10.1561/1500000090>

Psychology-informed Recommender Systems

Elisabeth Lex¹, Dominik Kowald², Paul Seitlinger³, Thi Ngoc Trang Tran¹, Alexander Felfernig¹ and Markus Schedl⁴

¹ Graz University of Technology; elisabeth.lex@tugraz.at

² Know-Center GmbH; dkowald@know-center.at

³ Tallinn University; pseiti@tlu.ee

¹ Graz University of Technology; ttrang@ist.tugraz.at

¹ Graz University of Technology; alexander.felfernig@ist.tugraz.at

⁴ Johannes Kepler University Linz and Linz Institute of Technology; markus.schedl@jku.at

ABSTRACT

Personalized recommender systems have become indispensable in today’s online world. Most of today’s recommendation algorithms are data-driven and based on behavioral data. While such systems can produce useful recommendations, they are often uninterpretable, black-box models, which do not incorporate the underlying cognitive reasons for user behavior in the algorithms’ design. The aim of this survey is to present a thorough review of the state of the art of recommender systems that leverage psychological constructs and theories to model and predict user behavior and improve the recommendation process. We call such systems *psychology-informed recommender systems*. The survey identifies three categories of psychology-informed recommender systems: *cognition-inspired*, *personality-aware*, and *affect-aware* recommender systems. Moreover, for each category, we highlight domains, in which psychological theory plays a key role and is therefore considered in the recommendation process. As recommender systems are fundamental tools to support human decision making, we also discuss selected decision-psychological phenomena that impact the interaction between a user and a recommender. Besides, we discuss related work that investigates the evaluation of recommender systems from the user perspective and highlight user-centric evaluation frameworks. We discuss potential research tasks for future work at the end of this survey.

1

Introduction

1.1 Motivation

In the past twenty years, research on recommender systems has emerged as a growing field within computer science (Ricci *et al.*, 2011). The emergence of online marketplaces, online social networks, online collaboration platforms, and online social information systems (Caverlee *et al.*, 2010) has created a need to support users with recommendations to help them cope with the increase of information and items online (Liu *et al.*, 2014).

A large amount of work exists that has tackled recommender systems research from a broad range of perspectives. Resources like the *Recommender Systems Handbook* (Ricci *et al.*, 2015) or *Recommender systems: An Introduction* (Jannach *et al.*, 2010) give a comprehensive overview of the field. So do review articles such as (Jannach *et al.*, 2012). Recent surveys provide a concise overview of explainable recommendations (Zhang, Chen, *et al.*, 2020), deep learning in recommender systems (Xu *et al.*, 2020), adversarial recommender systems (Deldjoo *et al.*, 2021b) or conversational recommender systems (Jannach *et al.*, 2020).

Early work on recommender systems was motivated by the observation that humans tend to base their decisions on the recommendations provided by their social surrounding (Ricci *et al.*, 2011). Correspondingly, the first algorithms developed as recommender systems aimed to mimic this behavior (Resnick and Varian, 1997; Ricci *et al.*, 2011). In the early 2000s, the use of psychological models in recommender systems research has gained traction. Pioneering work was carried out by Gustavo Gonzalez, Timo Saari, and Judith Masthoff, which exploited the psychological characteristics of users to improve the recommendation process. To that end, Gonzales *et al.* (González *et al.*, 2002; González *et al.*, 2004) considered emotional aspects of the user to generate personalized recommendations. Saari *et al.* (Saari *et al.*, 2004b; Turpeinen and Saari, 2004; Saari *et al.*, 2004a; Saari *et al.*, 2004a; Saari *et al.*, 2005) designed recommender systems that incorporate a user's emotion and attention, as well as other related constructs, to deliver recommendations (Nunes, 2008). Masthoff *et al.* (Masthoff, 2004b; Masthoff, 2004a; Masthoff, 2005; Masthoff and Gatt, 2006), assessed the user satisfaction of individual users and predicted group satisfaction when recommending sequences of items to user groups. Their intuition was that the first few recommendations in a list of recommendations influence the mood of the user. That mood, in turn, can impact the views the user has about the next items in the recommendation list (Nunes, 2008). Felfernig *et al.* (2007) used insights from decision psychology to gain a deeper understanding of online buyer behavior and to

improve knowledge-based recommender systems.

In the present survey article, we provide a review of research strands in the recommender systems community that enrich data-driven recommendation techniques with psychological constructs to design or improve recommender systems. We call such systems *psychology-informed recommender systems*.

This survey is organized as follows. We first give an introduction into common recommender systems methods in Section 1.2, and then, in Section 1.4, briefly describe our survey method and research scope. Next, in Section 2, we review related work on psychology-informed recommender systems, which we categorize into *cognition-inspired*, *personality-aware*, and *affect-aware* recommender systems. Also, in Section 3, we review works that investigate various decision-psychological phenomena that come into play when users interact with a recommender system. Besides, in Section 4, we discuss works that investigate recommender systems' evaluation from the user perspective. We conclude in Section 5 with key findings and possible directions for future work.

1.2 Main Approaches to Recommender Systems

The most prominent recommendation approaches are collaborative filtering (CF), content-based filtering (CBF), hybrid combinations of both (Ricci *et al.*, 2015), as well as knowledge-based recommender systems (Burke, 2000b). CF (Schafer *et al.*, 2007) exploits interactions between users and items such as ratings and creates a user-item matrix that is then used to predict missing ratings for pairs of users and items. CF then recommends the items with the highest predicted ratings, with which the target user has not yet interacted. One can distinguish between *model-based CF* and *memory-based CF* (Koren and Bell, 2015). In the case of *model-based CF* (Aggarwal, 2016), the algorithm first projects users and items into a low-dimensional space and then, finds similar users/items in this space. In the case of *memory-based CF* (Sarwar *et al.*, 2001), CF computes similarities between users/items directly from the user-item matrix. *Memory-based CF* can be further divided into *user-based CF* and *item-based CF*, depending on whether recommendations are produced based on user or item similarity.

CBF exploits characteristic properties of items (e.g., movie genres) to recommend items with similar attributes as items the target user has liked in the past (Ricci *et al.*, 2015). For a recent overview of new trends in CBF, please refer to Lops *et al.*, 2019. Correspondingly, hybrid recommender systems (Burke, 2002) are, most commonly, a combination of collaborative and content-based methods. For example, when using CF in a cold-start scenario, a hybrid approach can incorporate CBF to predict items based on their features (Cremonesi *et al.*, 2011b; Ricci *et al.*, 2011).

In contrast to CF and CBF, knowledge-based recommender systems (Burke, 2000b) do not require a user history. Instead, they make use of pre-existing knowledge about the user and the application domain to reason about potentially relevant items. One can distinguish between two main types of knowledge-based recommender systems, namely, constraint-based recommender systems (Felfernig and Burke, 2008; Atas *et al.*, 2019) and case-based recommender systems (Lorenzi and Ricci, 2003; Burke, 2000a). In constraint-based recommender systems, explicitly defined constraints govern which items should be recommended to a user in a given context,

Name	URL	Comments
LKPy	https://github.com/lenskit/lkpy	Python; classical models
Surprise	https://github.com/NicolasHug/Surprise	Python; classical models
pyRecLab	https://github.com/gasevi/pyreclab	Python; classical models
LibRec	https://github.com/guoguibing/librec	Java; classical models
Elliot	https://github.com/sisinflab/elliot	Python; classical and deep models
NeuRec	https://github.com/wubinzu/NeuRec	Python; deep models
Spotlight	https://github.com/maciejkula/spotlight	Python; classical and deep models
Implicit	https://github.com/benfred/implicit	Python; for implicit-feedback datasets
TagRec	https://github.com/learning-layers/TagRec	Java; cognition-inspired and classical models

Table 1.1: Overview of selected software for recommender systems.

whereas the constraints refer to the user and/or the item domain. Case-based recommender systems are early examples of psychology-informed recommender systems, which model reasoning as primarily memory-based (Leake, 2015). In this paper, they are, therefore, reviewed in more detail (see Section 2.1.4).

1.3 Selected Recommender Systems Software and Datasets

To facilitate getting started with recommender systems experiments, we provide an overview of relevant resources. Tables 1.1 and 1.2 give a non-exhaustive list of software¹ (libraries and open-source code repositories) and datasets, respectively.² We focus on the most popular resources as well as on those that provide code and data relevant to psychology-informed recommendation.

1.4 Survey Method and Research Scope

For this survey, we investigated research articles that appeared in relevant publication outlets in the fields of computer science, psychology, and human-computer-interaction. Regarding the scope of our review, we focus on papers that describe algorithms, techniques, and systems that exploit psychological features of the user for improving the recommendation process (see Table 1.5, Table 1.6, Table 1.8, and Table 1.9). Also, we visualize the reviewed papers as a timeline in Table 1.3, and Table 1.4 to show the evolution of techniques over time. Please note that we split the timeline visualization into periods from 1885 to 2010 and 2011 to 2021 due to space constraints.

The identification of papers for our survey was done according to the following strategy. We first considered the proceedings and volumes of a set of relevant conference series (e.g., *User Modelling, Adaptation and Personalization*, *ACM Recommender Systems Conference*, *The Web Conference*, *ACM SIGIR Conference on Research and Development in Information Retrieval*, *ACM CHI Conference on Human Factors in*

¹See also https://github.com/grahamjenson/list_of_recommender_systems & <https://recommender-systems.com/resources/>

²GroupLens' list of datasets: <https://grouplens.org/datasets/>, Julian McAuley's list: <https://cseweb.ucsd.edu/~jmcauley/datasets.html>

Computing Systems, ACM Hypertext, IEEE/WIC/ACM International Conference on Web Intelligence) and journals (e.g., User Modeling and User-Adapted Interaction, Transaction on Intelligent Information Systems, Cognitive Science, Journal of Consumer Research, IEEE Transactions on Affective Computing, Computers in Human Behavior, Journal of Personality and Social Psychology, ACM Transactions on Intelligent Information Systems) for articles that fall into the above-described scope. Additionally, we used the keywords “psychology recommender systems”, “psychology informed recommender”, “cognition recommender”, “stereotypes recommender”, “case-based recommender”, “affective recommender”, “emotion recommender”, “personality recommender”, “decision making recommender”, “user-centric recommender”, “user evaluation recommender”, “user experience recommender”, “nudging recommender systems”, “persuasion recommender”, “cognitive dissonance recommender”, “interaction recommender”, and “interfaces recommender” to search for papers in Google Scholar. Using the resulting set of articles as a starting point, we followed the references of the retrieved articles to find additional papers.

A few survey works on the topic of psychological models in the context of recommendations already exist. When looking at these existing works, we find that some works on psychology-informed recommender systems are also summarized by Tkalcic and Chen (2015a) with respect to personality-based recommender systems, personality and learning styles (Graus and Ferwerda, 2019), and in (Tkalcic *et al.*, 2011) in terms of affective-based systems. Additionally, Buder and Schwind (2012) discuss personalized recommender systems as well as psychological theories and models that describe learning processes and mechanisms in educational contexts. They, however, focus only on learning as a domain. Yoo *et al.* (2012), and in earlier work, Gretzel and Fesenmaier (2006), discuss recommender systems and their persuasive role in decision-making processes; Felfernig *et al.* (2008b) outline persuasion in knowledge-based recommendation. These works also shed light on psychological constructs that play a role in persuasion, which corresponds to a mechanism that can be used in recommender systems to influence choices. For a detailed overview of persuasive recommender systems, please refer to Yoo *et al.* (2012). Jesse and Jannach (2021) review related work on nudging with recommender systems. They also discuss 58 psychological mechanisms that are described in the reviewed works. Pu *et al.* (2012) present a survey on evaluating recommender systems from the user perspective, including preference elicitation and refinement, presentation of recommendations, and user-centric evaluation frameworks. Also, the authors summarize the most important results in the form of design guidelines for effective recommender systems.

Explanations of algorithmic decisions made by artificial intelligence help making algorithms more transparent. The recent survey on explainable recommendations by Zhang, Chen, *et al.* (2020) discusses related work on explainable recommendation models. For an overview of the body of research on explanations in artificial intelligence in light of the social sciences, please refer to Miller (2019).

In Zhang, Chen, *et al.* (2020) explanations in recommender systems are related to cognitive science and human decision making. As the authors describe, humans sometimes decide using rational and careful reasoning, while in other cases, they first decide and find explanations for their decisions later. This is in line with the typical approaches to designing explainable recommendation models: either, such models

are already designed with transparency and explainability in mind, or post-hoc explanations are used to explain decisions made by black box models (Lipton, 2018; Miller, 2019). Tran *et al.* (2019a) and Tran *et al.* (2020) take into account findings from social choice theory, i.e., the study of collective choices that impact groups (Sen, 1986), to introduce explanations to increase fairness, consensus, and satisfaction of users with group recommendations.

Given the rich body of work on explainability in recommender systems, which is already presented in the survey by Zhang, Chen, *et al.* (2020), we do not focus on this topic in the paper at hand, instead refer the reader to Zhang, Chen, *et al.* (2020) as well as to the respective chapter in the recommender systems handbook by Tintarev and Masthoff (2015).

The field of group recommender systems also uses social psychology constructs to produce recommendations that are helpful for groups. In this paper, we touch upon them when we discuss relevant work on personality in group recommender systems. For an overview of group recommender systems and mechanisms to model group behavior, please also refer to Felfernig *et al.* (2018c) and Masthoff (2015).

Summing up, with this article, we aim to close the gap between a computer science perspective (in particular, a technical recommendation systems point of view) and a psychological perspective. We hope to appeal to researchers in the information retrieval and recommendation systems communities who want to delve deeper into the psychological foundations of recommendation systems research. In addition, we also address an audience with psychological background who strives to deepen their knowledge on how psychological constructs and models can be incorporated into recommendation systems. Please note that basic knowledge of recommendation systems and psychology is sufficient to understand the article.

Name	URL	Domain	Comments
MovieLens	https://grouplens.org/datasets/movielens	movie	ratings, tags
FilmTrust	https://grouplens.org/datasets/filmtrust/	movie	ratings, trust scores
Epinions, Ciao	https://www.cse.lehigh.edu/~tangjili/datasets/ciao/	movie	movie ratings, reviews, review ratings, trust scores
Personality 2018	https://grouplens.org/datasets/personality-2018/	movie	movie preferences, personality information, ratings (with timestamps)
Serendipity 2018	https://grouplens.org/datasets/serendipity-2018	movie	movie ratings (with timestamps), survey responses related to serendipity preferences
Million Song Dataset	http://millionsongdataset.com	music	listening events, tags, genres, lyrics
LFM-1b	http://www.cs.jhu.edu/datasets/LFM-1b	music	music listening events (with time stamps), tags, user demographics
Million Playlist Dataset	https://www.alicebox.com/challenge/opify-million-playlist-dataset-challenge	music	public user-generated playlists from Spotify
HetRec 2011	https://grouplens.org/datasets/hetrec-2011	social networking, social tagging systems	tag assignments, bookmarks, movie genres, movie genre assignments

Table 1.2: Overview of selected datasets for recommender systems.

	Cognition-inspired	Personality-aware	Affect-aware	Decision Making	User-centric Eval.
1885	Ebbinghaus (1885)		empty citation	empty citation	empty citation
1954	empty citation		empty citation	empty citation	Festinger (1964)
1957	empty citation		empty citation	empty citation	Dense and Kihlstrom (1957)
1960	empty citation		empty citation	empty citation	Chaitin and Cernitz (1960)
1967	Natalis (1967)		empty citation	empty citation	empty citation
1967	Anderson (1974)		empty citation	empty citation	empty citation
1974	Martin and Stang (1978)		empty citation	empty citation	Tversky and Kahneman (1974)
1975	Elaine Rich (1975)		empty citation	empty citation	empty citation
1979	Russell (1980) and Mehrabian (1980)		empty citation	empty citation	empty citation
1980	empty citation		empty citation	empty citation	empty citation
1981	empty citation		empty citation	empty citation	Tversky and Kahneman (1981)
1982	empty citation		empty citation	empty citation	Huber <i>et al.</i> (1982)
1984	Ingvorsen (1984)		empty citation	empty citation	empty citation
1990	Ormerod (1990)		empty citation	empty citation	McCloskey <i>et al.</i> (1984)
1992	Borland <i>et al.</i> (2012)		empty citation	empty citation	empty citation
1993	Barling (1993), Flynn (1994)		empty citation	empty citation	Tversky and Kahneman (1992)
1994	Lamont (1994), Flynn (1994)		empty citation	empty citation	empty citation
1995	Costa and McCrae (1995)		empty citation	empty citation	empty citation
1997	Anderson <i>et al.</i> (1997)		empty citation	empty citation	empty citation
1999	Burke (1999)		empty citation	empty citation	Bordachevsky and Neunenschwander (1999)
2000	empty citation		empty citation	empty citation	Herbecker <i>et al.</i> (2000)
2001	Ricci and Werthner (2001)		empty citation	empty citation	Allen and Yen (2001)
2002	Ricci <i>et al.</i> (2002), Agnizoli <i>et al.</i> (2002)		empty citation	empty citation	Poggie (2002) and Swearingen and Shaha (2002)
2003	empty citation		empty citation	empty citation	McNee <i>et al.</i> (2003), and Cosley <i>et al.</i> (2003)
2004	empty citation		empty citation	empty citation	Briñolker <i>et al.</i> (2004)
2005	Anderson (2005)		empty citation	empty citation	Zhang <i>et al.</i> (2005), and Ling <i>et al.</i> (2005)
2006	Ricci <i>et al.</i> (2006)		empty citation	empty citation	Pu and Chen (2006)
2007	Furn <i>et al.</i> (2007), Rutledge-awiller <i>et al.</i> (2007), and El-Ghoulko <i>et al.</i> (2007)		empty citation	empty citation	Feltenine <i>et al.</i> (2007)
2008	Rutledge-Taylor <i>et al.</i> (2008), and Prins <i>et al.</i> (2008), and Maanen and Marewski (2009), and Wong (2009), and Yang and Wong (2009), and Quijano-Sánchez <i>et al.</i> (2010)		empty citation	empty citation	O'Brien and Toms (2008) and Feltening <i>et al.</i> (2008a)
2009			empty citation	empty citation	Crawall <i>et al.</i> (2008) and Thaler and Sunstein (2009)
2010	Fu <i>et al.</i> (2010), Fu and Kar-namalili (2010), and Yu and Li (2010)		empty citation	empty citation	Moizach and Schulz-Hardt (2010)
					Bollen <i>et al.</i> (2010), Chen and Pu (2010a), O'Brien and Toms (2010), and Nanon <i>et al.</i> (2010)

Table 1.3: Part I of a timeline visualization of the reviewed publications to depict the evolution of techniques (from 1885 to 2010); note that the earliest works are psychological papers that describe relevant effects).

Cognition-inspired	Personality-aware	Affect-aware	Decision Making	User-centric Eval.
2011 Blanco-Fernández <i>et al.</i> (2011)	Reutkoff <i>et al.</i> (2011) and Masthoff (2011)	Tkalčić <i>et al.</i> (2011)	Adomavicius <i>et al.</i> (2011), Zhang (2011), Maudí <i>et al.</i> (2011), Morawietz <i>et al.</i> (2011)	Shan and Gunawardana (2011), Hu and Pu (2011), Pu <i>et al.</i> (2011), Ekestrand <i>et al.</i> (2011), Yoo and Schwab <i>et al.</i> (2011), Yoo and Crenzel (2011), Kujimbenyi <i>et al.</i> (2011), Crenzel <i>et al.</i> (2011), Yannakakis and Huhun (2011), Tkalcic and Chen (2015a)
2012 Fu and Dong (2012); Psychology (2012); Boulen <i>et al.</i> (2012); Wang and Yang (2012); Belandri <i>et al.</i> (2012); and Doherty <i>et al.</i> (2012)	Tintarev and Masthoff (2012)	Konstan and Riedl (2012)	Yoo <i>et al.</i> (2012); Bettman <i>et al.</i> (1998), Teppan and Feltovich (2012); Murphy <i>et al.</i> (2012); Ranjith (2012); Bateman <i>et al.</i> (2012); and Smits (2013a); Adomavicius <i>et al.</i> (2013); and Thaler <i>et al.</i> (2013)	Ranjith (2012), Bateman <i>et al.</i> (2012), Chen <i>et al.</i> (2013a), Adomavicius <i>et al.</i> (2013), Konstan and Riedl (2012), Kujimbenyi and Schwab <i>et al.</i> (2012), Pu <i>et al.</i> (2012), Kujimbenyi and Crenzel <i>et al.</i> (2012), Yannakakis and Huhun (2011), Tkalcic and Chen (2015a)
2013 Sakater-nir <i>et al.</i> (2013), Kowald <i>et al.</i> (2013), Seitzinger <i>et al.</i> (2013), and Fahlberg (1993)	Golbeck and Norris (2013); Chen <i>et al.</i> (2013b); Wu <i>et al.</i> (2013); Tintarev <i>et al.</i> (2013), and Cantador <i>et al.</i> (2013)	Zhang (2013)	empty citation	empty citation
2014 Beel <i>et al.</i> (2014); Kowald <i>et al.</i> (2014); Zhao <i>et al.</i> (2014); Masterson <i>et al.</i> (2014); and Chavatridge <i>et al.</i> (2014)	Yoo <i>et al.</i> (2012); Bettman <i>et al.</i> (1998), Teppan and Feltovich (2012); Murphy <i>et al.</i> (2012); Ranjith (2012); Bateman <i>et al.</i> (2012), Chen <i>et al.</i> (2013a), Adomavicius <i>et al.</i> (2013), Konstan and Riedl (2012), Kujimbenyi and Schwab <i>et al.</i> (2012), Pu <i>et al.</i> (2012), Kujimbenyi and Crenzel <i>et al.</i> (2012), Yannakakis and Huhun (2011), Tkalcic and Chen (2015a)	empty citation	empty citation	empty citation
2015 Ren (2015); Kowald and Lex (2015); Musso <i>et al.</i> (2015); Bouabali and Choffi (2015); Mihannimad <i>et al.</i> (2015); Beel and Langer (2015); and Beel <i>et al.</i> (2015)	Orellana-Rodriguez <i>et al.</i> (2015) and Deng <i>et al.</i> (2015)	Jamison <i>et al.</i> (2015), Kartini <i>et al.</i> (2015), Stettlinger <i>et al.</i> (2015b), Tintarev <i>et al.</i> (2015), and Sunstein (2015)	empty citation	empty citation
2016 Seitlinger and Ley (2016); Jones (2016); Trattner <i>et al.</i> (2016); Kováč and Ševčík (2016); Sharkey (2016); and Cervone (2016)	Karumur <i>et al.</i> (2016), Fernandes-Tobias <i>et al.</i> (2016), and Roassi (2016)	empty citation	empty citation	empty citation
2017 Beel (2017); Kowald <i>et al.</i> (2017b), and Kopinski <i>et al.</i> (2017b); Harvey <i>et al.</i> (2016); Kopp (2016); and Mozer (2016); and Lindley (2016); and Tintarev (2017); and Doherty <i>et al.</i> (2017)	Fewverda <i>et al.</i> (2017b), Fewverda <i>et al.</i> (2017c), Naïmpanits and Tjoelstis (2017), and Doherty <i>et al.</i> (2017)	Piazza <i>et al.</i> (2017), Schell <i>et al.</i> (2018), and Ravi and Väistövaara (2018)	Joachims <i>et al.</i> (2017), Elswetter <i>et al.</i> (2017), and Ravi and Väistövaara (2017)	Kamisskas and Bridge (2016), Ekstrand and Wilmesen (2016), and Wilmesen <i>et al.</i> (2016)
2018 AlRossais and Kudenko (2018); Thaker <i>et al.</i> (2018); Yagci <i>et al.</i> (2018); Farrell and Lewandowsky (2018); and Chintal and Shinde (2018)	Nguyen <i>et al.</i> (2018), Karumur <i>et al.</i> (2018), Wu <i>et al.</i> (2018), Liu <i>et al.</i> (2018), Asabere <i>et al.</i> (2018), Aksel <i>et al.</i> (2018), and Henning <i>et al.</i> (2018)	Ayata <i>et al.</i> (2018)	Joachims <i>et al.</i> (2017), Elswetter <i>et al.</i> (2017), and Hertwig and Grimeyer (2017), Schneider <i>et al.</i> (2018), and Grimeyer and Jugovac (2018)	Herlocker <i>et al.</i> (2017), Jugovac and Jannach (2017), and Meike and Pothoff (2017)
2019 Aragoneses <i>et al.</i> (2019); Zhou <i>et al.</i> (2019); Yang <i>et al.</i> (2019); and Nguyen <i>et al.</i> (2019)	Ying (2019); Sertikcan <i>et al.</i> (2019); and Zhang <i>et al.</i> (2019)	Mizgajski and Morzy (2019)	Jugovac <i>et al.</i> (2018), Tran <i>et al.</i> (2018), and Grimeyer and Jugovac (2018)	Jugovac <i>et al.</i> (2018)
2020 Kahans (2020); Lox <i>et al.</i> (2020); Conteras and Salamó (2020); and Ciulli <i>et al.</i> (2020)	Beheshti <i>et al.</i> (2020)	Perloff (2020)	empty citation	empty citation
2021 empty citation	empty citation	Jesse and Jannach (2021)	empty citation	Orloff <i>et al.</i> (2021)

Table 1.4: Part II of a timeline visualization of the reviewed publications to depict the evolution of techniques (from 2011 to 2021).

Cognition	Sec.	References
Stereotypes	2.1	Elaine Rich, 1979; Rich, 1989; Blanco-Fernández <i>et al.</i> , 2011; Beel <i>et al.</i> , 2014; Beel and Langer, 2015; Beel <i>et al.</i> , 2015; Beel, 2015; ALRossais and Kudenko, 2018; ALRossais, 2018
Cogn. Models	2.1.1	Anderson, 2005; Fum <i>et al.</i> , 2007; Farrell and Lewandowsky, 2018; Neisser, 1967; Ormerod, 1990; Psychology, 2012; Jones, 2016; Glushko <i>et al.</i> , 2008; Fu, 2008; Fu <i>et al.</i> , 2010; Fu and Kannampallil, 2010; Fu and Dong, 2012; Anderson <i>et al.</i> , 1997
Memory	2.1.2	Seitlinger and Ley, 2016; Kahana, 2020; Ingwersen, 1984; Rutledge-Taylor and West, 2007; Rutledge-Taylor <i>et al.</i> , 2008; Anderson, 1974; Bollen <i>et al.</i> , 2012; Matlin and Stang, 1978; Ebbinghaus, 1885; Ebbinghaus, 2013; Yu and Li, 2010; Ren, 2015; Chmiel and Schubert, 2018; Yang <i>et al.</i> , 2019; Sabatermir <i>et al.</i> , 2013; Maanen and Marewski, 2009; Kowald <i>et al.</i> , 2014; Trattner <i>et al.</i> , 2016; Kowald <i>et al.</i> , 2013; Kowald <i>et al.</i> , 2017b; Kowald and Lex, 2016; Kowald and Lex, 2015; Stanley and Byrne, 2016; Kowald <i>et al.</i> , 2020a; Kopeinik <i>et al.</i> , 2016; Kopeinik <i>et al.</i> , 2017b; Kowald <i>et al.</i> , 2019; Lex <i>et al.</i> , 2020; Zhao <i>et al.</i> , 2014; Missier, 2014; Schnabel <i>et al.</i> , 2016; Elsweiler <i>et al.</i> , 2007; Harvey <i>et al.</i> , 2016; Doherty <i>et al.</i> , 2012; Gemmell <i>et al.</i> , 2002; Lamming and Flynn, 1994
Attention	2.1.3	Seitlinger <i>et al.</i> , 2013; Kowald <i>et al.</i> , 2013; Kopeinik <i>et al.</i> , 2017a
CBR	2.1.4	Hammond, 2012; Kolodner, 2014; Riesbeck and Schank, 2013; Kolodner, 1992; Tversky, 1977; Burke <i>et al.</i> , 1996; Burke, 1999; Ricci and Werthner, 2001; Ricci <i>et al.</i> , 2002; Ricci <i>et al.</i> , 2006; Aguzzoli <i>et al.</i> , 2002; Gong, 2009; Yang and Wang, 2009; Wang and Yang, 2012; Musto <i>et al.</i> , 2015; Bousbahi and Chorfi, 2015; McSherry, 2005; Sharma and Ray, 2016; Muhammad <i>et al.</i> , 2015; Jorro-Aragoneses <i>et al.</i> , 2019; Pu <i>et al.</i> , 2012; McGinty and Reilly, 2011; Contreras and Salamó, 2020; Contreras and Salamó, 2020; Güell <i>et al.</i> , 2020
Competence	2.1.5	Fehling, 1993; Bellandi <i>et al.</i> , 2012; Chavarriaga <i>et al.</i> , 2014; Prins <i>et al.</i> , 2008; Yago <i>et al.</i> , 2018; Mozer and Lindsey, 2016; Thaker <i>et al.</i> , 2018

Table 1.5: Overview of surveyed papers that implement cognitive models to design and improve recommendation techniques.

Personality-aware Rec. Sys.	Sec.	References
Personality	2.2	Tkalcic and Chen, 2015a ; Ferwerda <i>et al.</i> , 2017b ; Golbeck and Norris, 2013 ; Rentfrow <i>et al.</i> , 2011 ; Chen <i>et al.</i> , 2013b ; Wu <i>et al.</i> , 2013 ; Nguyen <i>et al.</i> , 2018 ; Karumur <i>et al.</i> , 2018 ; Karumur <i>et al.</i> , 2016
Personality Elicitation	2.2.1	McCrae and John, 1992 ; Thomas, 1992 ; Felfernig <i>et al.</i> , 2018d ; Holland, 1997 ; Bologna <i>et al.</i> , 2013 ; Stewart, 2011 ; Konert <i>et al.</i> , 2013 ; Paiva <i>et al.</i> , 2015 ; Goldberg <i>et al.</i> , 2006 ; Gosling <i>et al.</i> , 2003 ; John and Srivastava, 1999 ; Berkovsky <i>et al.</i> , 2019 ; Wu <i>et al.</i> , 2019 ; Ferwerda and Tkalcic, 2018 ; Golbeck <i>et al.</i> , 2011a ; Golbeck <i>et al.</i> , 2011b ; Golbeck, 2016
Personality Traits in RecSys	2.2.2	Asabere <i>et al.</i> , 2018 ; Yang and Huang, 2019 ; Adaji <i>et al.</i> , 2018 ; Nalmpantis and Tjortjis, 2017 ; Cantador <i>et al.</i> , 2013 ; Gelli <i>et al.</i> , 2017 ; Tintarev <i>et al.</i> , 2013 ; Wu <i>et al.</i> , 2018 ; Ferwerda <i>et al.</i> , 2017a ; Lu and Tintarev, 2018 ; Fernandez-Tobias <i>et al.</i> , 2016 ; Beheshti <i>et al.</i> , 2020 ; Sertkan <i>et al.</i> , 2019
Personality in Group RecSys	2.2.3	Recio-Garcia <i>et al.</i> , 2009 ; Felfernig <i>et al.</i> , 2018a ; Masthoff, 2011 ; Quijano-Sanchez <i>et al.</i> , 2010 ; Rossi and Cervone, 2016 ; Costa and McCrae, 1995 ; Charness and Rabin, 2002 ; Delic <i>et al.</i> , 2017 ; Nguyen <i>et al.</i> , 2019

Table 1.6: Overview of our surveyed papers describing personality-aware recommendation algorithms and systems.

Affect-aware RecSys	Sec.	References
Affect	2.3	Shiv and Fedorikhin, 1999 ; Orellana-Rodriguez <i>et al.</i> , 2015 ; Piazza <i>et al.</i> , 2017 ; Ferwerda <i>et al.</i> , 2017b ; Golbeck and Norris, 2013 ; Rentfrow <i>et al.</i> , 2011 ; Chen <i>et al.</i> , 2013b ; Wu <i>et al.</i> , 2013 ; Mizgajski and Morzy, 2019 ; Schäfer, 2016 ; Schedl <i>et al.</i> , 2018 ; Zheng, 2013
Modeling Affect	2.3.1	Russell, 1980 ; Mehrabian, 1980 ; Fontaine <i>et al.</i> , 2007
Affect in RecSys	2.3.2	Tkalcic <i>et al.</i> , 2011 ; Ravi and Vairavasundaram, 2017 ; Deng <i>et al.</i> , 2015 ; Ayata <i>et al.</i> , 2018

Table 1.7: Overview of the surveyed papers describing affect-aware recommendation algorithms and systems.

Human Decision Making	Sec.	References
Decision Making	3	Yoo <i>et al.</i> , 2012; Chen <i>et al.</i> , 2013a; Bettman <i>et al.</i> , 1998; Jameson <i>et al.</i> , 2015; Adomavicius <i>et al.</i> , 2013; Tversky and Kahneman, 1974; Chapman and Johnson, 2002; Karimi <i>et al.</i> , 2015; Jugovac <i>et al.</i> , 2018
Decoy Items	3.1	Payne <i>et al.</i> , 1993; Huber <i>et al.</i> , 1982; Teppan and Felfernig, 2012; Teppan and Zanker, 2015
Serial Position Effects	3.2	Deese and Kaufman, 1957; Glanzer and Cunitz, 1966; Ranjith, 2012; Murphy <i>et al.</i> , 2012; Felfernig <i>et al.</i> , 2007; Schnabel <i>et al.</i> , 2016; Stettiner <i>et al.</i> , 2015a; Tran <i>et al.</i> , 2018; Hofmann <i>et al.</i> , 2014; Joachims <i>et al.</i> , 2017; Craswell <i>et al.</i> , 2008; Stettiner <i>et al.</i> , 2015b; Dyer, 2005
Framing	3.3	Tversky and Kahneman, 1981; Tversky and Kahneman, 1992; Mandl <i>et al.</i> , 2011
Anchor Effects	3.4	Mojzisch and Schulz-Hardt, 2010; Adomavicius <i>et al.</i> , 2011; Zhang, 2011; Köcher <i>et al.</i> , 2019; Adomavicius <i>et al.</i> , 2014; Felfernig <i>et al.</i> , 2018b
Nudging	3.5 & 3.6	Thaler and Sunstein, 2009; Thaler <i>et al.</i> , 2013; Tversky and Kahneman, 1974; Jesse and Jannach, 2021; Karlsen and Andersen, 2019; Caraban <i>et al.</i> , 2019; Elsweiler <i>et al.</i> , 2017; Esposito <i>et al.</i> , 2017; Turland <i>et al.</i> , 2015; Schneider <i>et al.</i> , 2018; Sunstein, 2015
Boosting	3.6	Grüne-Yanoff and Hertwig, 2016; Hertwig and Grüne-Yanoff, 2017; Grüne-Yanoff <i>et al.</i> , 2018; Zimmerman <i>et al.</i> , 2020; Ortloff <i>et al.</i> , 2021; Bateman <i>et al.</i> , 2012; Moraveji <i>et al.</i> , 2011

Table 1.8: Overview of the surveyed papers describing mechanisms of human decision making in light of recommender systems research.

User-centric Evaluation	Sec.	References
User-centric Evaluation	4.1	Ekstrand and Willemsen, 2016 ; Knijnenburg <i>et al.</i> , 2012a ; Mcnee <i>et al.</i> , 2006b ; Nalmpantis and Tjortjis, 2017 ; Chen and Pu, 2005 ; Konstan and Riedl, 2012 ; Xiao and Benbasat, 2007 ; Shin, 2020 ; Mcnee <i>et al.</i> , 2003 ; Ziegler <i>et al.</i> , 2005 ; O'Brien and Toms, 2008 ; Pu and Chen, 2006 ; Cosley <i>et al.</i> , 2003 ; O'Brien and Toms, 2010
Cognitive Dissonance	4.1.1	Festinger, 1954 ; Surendren and Bhuvaneswari, 2014 ; Schwind <i>et al.</i> , 2011 ; Kuan <i>et al.</i> , 2007 ; Schwind and Buder, 2014 ; Nguyen <i>et al.</i> , 2007
Persuasion	4.1.2	Fogg, 2002 ; Perloff, 2020 ; Meske and Potthoff, 2017 ; Yoo <i>et al.</i> , 2012 ; Gretzel and Fesenmaier, 2006 ; Jugovac <i>et al.</i> , 2018 ; Yoo and Gretzel, 2011 ; Nanou <i>et al.</i> , 2010 ; Cremonesi <i>et al.</i> , 2012 ; Felfernig <i>et al.</i> , 2008a ; Herlocker <i>et al.</i> , 2000 ; Tintarev and Masthoff, 2012 ; Berdichevsky and Neunenschwander, 1999 ; Smids, 2012
Interactions & Interfaces	4.1.3	Knijnenburg <i>et al.</i> , 2011 ; Knijnenburg and Willemsen, 2015 ; Bollen <i>et al.</i> , 2010 ; Chen and Pu, 2010b ; Chen and Pu, 2010a ; Hu and Pu, 2011 ; Ekstrand <i>et al.</i> , 2014 ; Jugovac and Jannach, 2017
Attitudes & Beliefs	4.1.4	Cremonesi <i>et al.</i> , 2011a ; Pu <i>et al.</i> , 2011 ; Swearingen and Sinha, 2002 ; Bollen <i>et al.</i> , 2010 ; Willemsen <i>et al.</i> , 2016 ; Jin <i>et al.</i> , 2019
User Study Design	4.2	Allen and Yen, 2001 ; McCroskey <i>et al.</i> , 1984 ; Yannakakis and Hallam, 2011 ; O'Brien and Toms, 2008 ; O'Brien and Toms, 2010 ; Goretzko <i>et al.</i> , 2019 ; Knijnenburg and Willemsen, 2015 ; Pu <i>et al.</i> , 2011 ; Knijnenburg <i>et al.</i> , 2012b ; Ullman and Bentler, 2003

Table 1.9: Overview of the surveyed papers describing research on user experience and designing user studies.

2

Psychology-informed Recommendation Approaches

In this chapter, we review three categories of psychology-informed recommender systems: (i) cognition-inspired, (ii) personality-aware, and (iii) affect-aware recommender systems.

2.1 Cognition-inspired Recommender Systems

Cognition-inspired recommender systems employ models from cognitive psychology to design and improve recommender systems. Cognitive psychology is a field of research within psychology that investigates human mental processes such as decision-making, memory, or attention. Early recommender systems research has extensively drawn on findings from cognitive psychology, among other disciplines (Adomavicius and Tuzhilin, 2005). In this respect, one of the earliest recommender systems was the Grundy system (Elaine Rich, 1979; Rich, 1989) that grouped users into *stereotypes* to create book recommendations. Stereotype-based recommender systems produce recommendations based on generalizing assumptions about users, such as that computer scientists like science fiction books and historians like biographies (Beel *et al.*, 2017). The underlying psychological principle of stereotypes is the *representativeness heuristic* by Kahneman and Tversky (1972), which people apply when making decisions under uncertainty. It is a mental shortcut that people use when assessing if an object belongs to a specific category. They make this decision based on how representative they think the object is for a category.

In the Grundy system, users described their interests based on adjectives, which were then grouped into stereotypes. The psychological literature describes stereotypes as a form of categorization that humans apply to reduce complexity. Using stereotyping, humans group others based on common characteristics. For an overview of the cognitive mechanisms behind stereotyping, please refer to (Hamilton, 1979; Hamilton, 2015). Please note stereotyping is a trivial application of psychological principles to model users.

Later work employed stereotypes in a library reference manager system to produce book recommendations (Beel *et al.*, 2014; Beel and Langer, 2015) and in (Beel *et al.*, 2015; Beel, 2015) to recommend research papers to researchers at different stages of their academic career. In the latter case, stereotypes serve as a fallback mechanism when classic approaches such as collaborative filtering cannot deliver

recommendations, e.g., in cold-start scenarios. Blanco-Fernández *et al.* (2011) use consumption stereotypes in a knowledge-based recommender systems. Recent work by Al-Rossais and Kudenko (ALRossais and Kudenko, 2018; ALRossais, 2018) performs a comparative analysis of the performance of stereotype-based item modeling and non-stereotype-based item modeling. Specifically, they evaluate the efficacy of two stereotype-based recommendation approaches: First, they create user-based stereotypes using demographic data such as age and gender, and second, item-based stereotypes based on user preferences. They find that incorporating stereotypes can improve recommendation accuracy and that stereotypes can help with the new item problem, i.e., an item comes to the system for which no interactions are available. However, the authors also note that the creation of stereotypes is labor-intensive, especially in the case of manually created stereotypes. While stereotypes are a simple technique to model users, in the remainder of this paper, we review works that exploit more complex psychological constructs in recommender systems research.

In the following, we first briefly outline theories of cognitive processes. Subsequently, we review works which use computational cognitive models to generate and improve personalized recommendations.

2.1.1 Computational Modeling of Cognitive Processes

Cognitive processes and cognition are typically studied in cognitive science, a discipline in which researchers from neuroscience, artificial intelligence, and cognitive psychology aim to understand the functioning of the mind (Anderson, 2005). Cognitive scientists have developed a broad range of empirical methods to study cognition (Fum *et al.*, 2007). The predominant empirical approach is to conduct experiments and analyze behavioral data using statistical models from mathematical psychology, whose parameters represent cognitive constructs. A prominent example is the power law of forgetting (Anderson *et al.*, 1997), which models the rate at which the activation of memory units decays in time.

An increasingly popular technique is cognitive-computational modeling (Farrell and Lewandowsky, 2018) – an attempt to specify cognitive assumptions and to simulate parts of the human mind through computable models (Neisser, 1967; Ormerod, 1990; Psychology, 2012).

In recent times, cognitive-computational modeling also allowed to complement experimental studies with more data-driven approaches, which, e.g., make use of large-scale datasets of social information systems (e.g., (Jones, 2016)). Corresponding artifacts within these systems, such as tagged bookmarks, can be interpreted as manifestations of cognitive processes (e.g., categorization of Web resources and evolving information needs) and used to test theories of human cognition (e.g., (Glushko *et al.*, 2008)). Illustrative examples can be found in the studies of Fu and colleagues (e.g., (Fu, 2008; Fu *et al.*, 2010; Fu and Kannampallil, 2010; Fu and Dong, 2012)), who draw on the cognitive architecture ACT-R (Anderson *et al.*, 1997) (see below) to perform theory-guided analyses and simulations of users' tagging behavior in social media, resulting in a socio-cognitive user model of social tagging (Fu, 2008; Fu and Dong, 2012).

2.1.2 Cognitive Models of Memory

Memory is a fundamental process of human cognition that supports goal-directed interactions with our physical and social environment (Seitlinger and Ley, 2016). The cognitive process memory enables the encoding and storing of information in memory structures, i.e., short-term, working memory, and long-term memory, so that it can be later retrieved. When information is recorded into memory (i.e., encoded) it is bound to temporal and spatial context information in order to later enable a context-guided search of memory content (i.e., the process of controlled retrieval) (Kahana, 2020). This makes memory processes closely related to research problems in Information Retrieval (Ingwersen, 1984) and Recommender Systems. In the following, we provide a number of examples where recommender systems have been inspired or motivated by memory models.

Memory models have been used in recommender systems in various forms. Rutledge et al. (Rutledge-Taylor and West, 2007; Rutledge-Taylor *et al.*, 2008) propose a recommender system that is based on a cognitive model of human long-term memory, i.e., dynamically structured holographic memory (DSHM) (Rutledge-Taylor and West, 2007), to resemble how a human expert makes recommendations. This system can model various human memory effects such as the fan effect (Anderson, 1974), i.e., recognition times for a concept increases as more information is available about the concept. Bollen *et al.* (2012) exploit positivity effects from human memory theory to investigate temporal dynamics of ratings in recommender systems. According to the psychological literature, memories become more positive over time (Matlin and Stang, 1978). In an offline study, the authors find evidence for the existence of the positivity effect in ratings, i.e., movies receive higher ratings as time between release date and rating date increases. However, a corresponding user study shows a decline in rating score when movies were rated in a larger interval between watching and rating.

Another memory model from psychology, the Ebbinghaus forgetting curve (Ebbinghaus, 2013) is used to model changes in the interests of users. The Ebbinghaus forgetting curve is a psychological theory from 1880 that describes the decrease in ability of the human brain to retain memory over time. In recommender systems research, the curve has been used in several works (e.g., (Yu and Li, 2010; Ren, 2015; Chmiel and Schubert, 2018; Yang *et al.*, 2019)) to account for shifts in user interests by weighting the user feedback (e.g., ratings) using a nonlinear, time-based memory decay function. Yu and Li (2010) and Ren (2015) utilize the curve to design a novel collaborative filtering algorithm that accounts for shifts in user interests. Chmiel and Schubert (2018) use the Ebbinghaus forgetting curve to model drifts in user preferences in a music recommender system. Yang *et al.* (2019) use it to derive item embeddings in a collaborative filtering approach. They use the curve to divide user preferences into long-term and short-term preferences where recently rated items are weighted higher than dated items.

Models of human memory are sometimes part of broader *cognitive architectures*, which aim to draw a more holistic picture of how different cognitive domains work together to generate emergent phenomena, such as a coherent thought. For an overview of cognitive architectures, please refer to Chong *et al.* (2007). Sabater-mir *et al.* (2013) use the cognitive architecture Belief/Desire/Intention (BDI) as an intermediate between recommenders and their users. The cognitive architecture

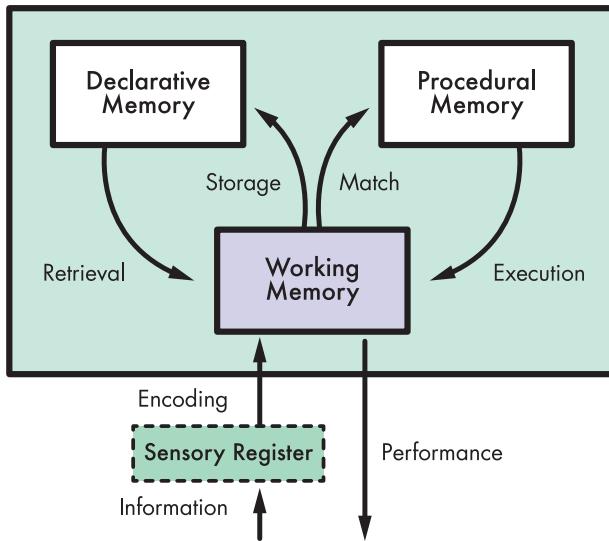


Figure 2.1: Schematic illustration of ACT-R (Kowald *et al.*, 2020a). Please note that the activation equation of the declarative memory module is used in a variety of recommender systems.

ACT-R (short for adaptive control of thought-rational) (Anderson *et al.*, 1997) has been employed in the context of recommender systems in several works (Maanen and Marewski, 2009; Kowald *et al.*, 2014; Trattner *et al.*, 2016; Kowald *et al.*, 2013; Kowald *et al.*, 2017b; Kowald and Lex, 2016; Stanley and Byrne, 2016)). ACT-R defines and formalizes the basic cognitive operations of the human mind.

Figure 2.1 depicts the main architecture of ACT-R. As illustrated in the figure, ACT-R differentiates between short-term memory modules, such as the working memory module, and long-term memory modules, such as the declarative and procedural memory modules. Using a sensory register (i.e., the ultra-short-term memory), the encoded information is passed to the short-term working memory module, which interacts with the long-term memory modules. In the case of the declarative memory, the encoded information can be stored, and already stored information can be retrieved. In the case of the procedural memory, the information can be matched against stored rules that can lead to actions (Kowald *et al.*, 2020a). Thus, declarative memory holds factual knowledge (e.g., what something is), and procedural memory consists of sequences of actions (e.g., how to do something). Most works that employ ACT-R in the context of recommender systems focus on the declarative part, which contains the activation equation of human memory. The activation equation determines the usefulness, i.e., the activation level of a memory unit (i.e., in the case of a recommender system, a candidate item) for a user in the current context.

According to ACT-R, the probability that a piece of information (i.e., a memory unit) will be needed to achieve a processing goal, i.e., will be *activated*, depends on its usefulness in the current context as well as a human's prior exposure to

this information. This prior exposure can be quantified by two factors: recency and frequency of usage. In addition, the current context in which the information occurs also contributes to its activation. All factors are modeled using ACT-R's activation equation, as given in Equation 2.1.

$$A_i = B_i + \sum_j W_j \cdot S_{j,i} \quad (2.1)$$

where A_i denotes the activation level of a memory unit i , B_i is the base-level activation of i , j is a cue in the current semantic context, W_j denotes the weighting of j , and $S_{j,i}$ is the strength of activation between j and i . B_i can be computed via the base-level learning (BLL) equation of ACT-R, i.e.:

$$B_i = \ln \left(\sum_{j=1}^n t_j^{-d} \right) \quad (2.2)$$

where n is the number of times i was activated in the past, t_j is the time since the j^{th} activation of i and d accounts of the time-based decay of activation in memory.

The activation equation of ACT-R has been exploited in several recommender systems. Maanen and Marewski (2009) use it to provide researchers at scientific conferences with recommendations of which talk to attend. Here, the recommender system mimics a researcher's memory since it recommends a talk if words in the talk's abstract have occurred recently and frequently in the scientist's work. Kowald *et al.* (2017b) use the equation to model and explain how Twitter users apply hashtags. They find that almost two-thirds of Twitter users in their datasets reuse their hashtags or social hashtags (i.e., from their friends' network), following a time-based decay in the form of a power-law function, in line with Equation 2.2. Based on these findings, they introduce a novel hashtag recommendation approach that adapts the equation to account for individual and social hashtag reuse and which ranks a user's hashtags and the ones of her friends based on frequency and recency. In other works, Kowald *et al.* (Kowald *et al.*, 2014; Kowald and Lex, 2016; Kowald and Lex, 2015) and Trattner *et al.* (2016) use the BLL equation (i.e., Equation 2.2) to model tag reuse processes and to recommend items in social tagging systems. Please note that these implementations of the BLL equation are available in the open source recommender systems framework *TagRec* (Kowald *et al.*, 2017a). Stanley and Byrne (2016) combine the equation with a random permutation vector-based model to describe past tagging behavior in StackOverflow and Twitter. Kopeinik *et al.* (2016) use it to recommend learning resources and support collaborative learning with tag recommendations (Kopeinik *et al.*, 2017b). Besides, in (Kowald *et al.*, 2019; Lex *et al.*, 2020), the activation equation is utilized to model music listening behavior and recommend artists and genres, respectively. The latter works show that the resulting computational model can alleviate popularity bias in music recommender systems (Kowald *et al.*, 2020b). Please note that the algorithms based on the activation equation to model music listening behavior are available in the open-source recommender systems framework *TagRec* (Kowald *et al.*, 2017a). Furthermore, Zhao *et al.* (2014) use the activation equation to produce context-aware recommendations for mobile applications as they combine frequency and recency of application use into contextual information.

Finally, recommender systems can support memory processes, as is described in (Missier, 2014). Here, Schnabel *et al.* (2016) propose to support a user's short-term memory by creating a digital short-term memory in the form of shortlists, which contain the items a user is currently considering. Items on the shortlist represent implicit feedback that is exploited to generate additional training data for a recommender system. Additionally, Elsweiler *et al.* (2007) relate the task of supporting memory in retrieving objects to recovering from memory lapses. They show that building upon research on how people recover from memory lapses can help to create better personal information management tools. Other works suggest augmenting human memory via providing documentation of events gathered from external tools such as wearable sensors or cameras (Harvey *et al.*, 2016; Doherty *et al.*, 2012). Related to this, in (Gemmell *et al.*, 2002), the *MyLifeBits* project is presented, which aims to fulfill Vannevar Bush's Memex vision to generate a system that is capable of reminding users of their stored "bits" (e.g., documents or images). A similar initiative, the *Forget-me-not* project, was even already introduced in 1994 (Lamming and Flynn, 1994). Interestingly, the authors state the importance of context for retrieving memory cues, which is in line with the recommendation algorithms mimicking human memory access that have been proposed decades later (e.g., by Kowald *et al.* (2017b)).

2.1.3 Cognitive Models of Attention

Attention is a mechanism to selectively process information in an environment in the face of distraction. Attention is dynamic in nature and hence typically modeled using connectionist models. Connectionism is a research strand in cognitive science, which uses artificial neural networks to study cognition and to model cognitive processes (Buckner and Garson, 2019). In this vein, Seitlinger *et al.* (2013) use the connectionist human memory simulation model ALCOVE (Kruschke, 1992) to implement a novel tag recommendation algorithm termed 3Layers. Kowald *et al.* (2013) enhance the 3Layers algorithm with recency effects by combining it with the BLL equation mentioned before. Another connectionist model is used by Kopeinik *et al.* (2017a), who apply SUSTAIN (Love *et al.*, 2004), a connectionist model of human category learning and successor of ALCOVE, to recommend resources that fit to a user's current attentional focus. Please note that the resulting resource recommendation algorithm is publicly available in the open-source TagRec framework (Kowald *et al.*, 2017a).

The use of memory or attention mechanisms in deep learning-based recommender systems (see e.g. (Zheng *et al.*, 2019)) has gained traction in recent years. To the best of our knowledge, such works typically do not discuss underlying psychological constructs. Therefore, in the present study, such works are omitted. For an overview of deep learning-based recommender systems, see (Zhang *et al.*, 2019b), as well as the recent study by Xu *et al.* (2020).

2.1.4 Cognition and Case-based Reasoning Recommender Systems

Case-based recommender systems (Hammond, 2012; Kolodner, 2014; Riesbeck and Schank, 2013) employ case-based reasoning (CBR), a technique pioneered by cognitive

scientist Janet Kolodner (Kolodner, 1992; Kolodner, 2014) to produce recommendations. CBR is a technique where a reasoner remembers previous cases that are similar to the current case and uses them to solve new problems (Kolodner, 1992). Such systems constitute early examples of psychology-informed recommender systems as they employ a problem solving architecture designed by psychologists. The similarity metrics used by CBR systems were inspired by works in psychology on the basic features of similarity. Here, the similarity between two items is determined based on their common and distinctive features (Tversky, 1977). Since CBR recommender systems are based on learning from previous experiences, they require a knowledge base that contains well-represented examples (Burke *et al.*, 1996).

CBR research examines the CBR process both as a model of human cognition and as an approach to build intelligent systems (Leake, 2001). In the context of recommender systems research, Burke employs CBR to generate recommendations in an e-commerce setting (Burke, 1999), and in Burke *et al.* (1996) to produce restaurant recommendations. Ricci *et al.* (Ricci and Werthner, 2001; Ricci *et al.*, 2002; Ricci *et al.*, 2006) utilize CBR in the domain of travel recommendations. Aguzzoli *et al.* (2002) combine CBR with CF to produce music recommendations, similar to Gong (2009), who combines CBR with item-based CF by first using CBR to fill missing entries in the user-item ratings matrix and then predicting items using CF. Yang and Wang (2009) designed an approach based on CBR to assist project managers in constructing new project plans based on previous projects. Wang and Yang (2012) introduce an extension to CBR to enable a hierarchical problem representation. Their approach considers multiple decision objectives on each level of hierarchical, multiple-level decision criteria; thus, problems can be identified more precisely. Musto *et al.* (2015) employ CBR to recommend personalized investment portfolios as an assisting tool to financial advisors. Bousbahi and Chorfi (2015) implement a CBR-based recommendation approach to assist learners in finding massive open online courses (MOOCs) that meet their personal interests.

CBR has furthermore been used to design explanation strategies for recommendations; see the respective chapter in the Recommender Systems Handbook for a concise overview of explanations for recommender systems (Tintarev and Masthoff, 2015); the review presented by Doyle *et al.* (2003) details the use of explanations in knowledge-based systems. McSherry (2005) explain recommendations along with the difference between query and case descriptions, whereas the query represents the user preferences. Sharma and Ray (2016) select the attribute with the highest weight in the similarity metric to find the similar cases that may be of interest to the user as an explanation of the recommendation.

Muhammad *et al.* (2015) describe a case-based recommender system for hotels whereas cases are extracted from the user-generated, textual reviews of users. In addition to cases, user profiles are created based on the reviews a user has submitted. Based on the profiles, a set of hotels are recommended and explanations for the candidates are produced and used to rank hotels. The explanations consist of hotel amenities enriched with sentiment extracted from opinions expressed in the reviews. Jorro-Aragoneses *et al.* (2019) introduce a CBR strategy to extract explanatory cases that are similar to recommended items, which are then used to interpret latent factors produced by matrix factorization recommendation algorithms.

Furthermore, critique-based recommender systems are a form of case-based

recommender systems (Pu *et al.*, 2012). Critique-based recommenders produce recommendations by creating a dialogue, in which recommendations are offered and users give feedback to the recommendations in the form of critiques. A large body of research exists on critique-based recommender systems; for an overview, please refer to the respective chapter in the Recommender Systems Handbook (McGinty and Reilly, 2011). In this field, recent work by Contreras and Salamó (2020) introduces a cognitive user preference model that incorporates an adaptive clustering process into the user model. The authors use this user model in a critique-based recommender system. Here, the cognitive user preference model is generated from interactions with the user and adapts its content to the evolving requirements of the user, which are defined by the user's critiques. Also in recent work, Güell *et al.* (2020) introduce a cognitive-based assistant for a critique-based recommender system, whose reasoning process when recommending products employs the same cognitively-inspired clustering algorithm as Contreras and Salamó (2020).

2.1.5 Competence-based Recommender Systems

Competence can be understood as the body of knowledge that is required to perform tasks in a particular domain (Fehling, 1993). In the context of recommender systems, competence is often used in learning and expert seeking scenarios. Bellandi *et al.* (2012) outline various design principles for competence-based recommender systems. The basis for such systems are competence profiles that help recommend expert advice or design teams. Chavarriaga *et al.* (2014) introduce a hybrid recommender system based on collaborative filtering and knowledge-based recommendations for students, which recommends activities and resources. The goal is to assist students achieve certain competence levels in the context of an online or blended course. Prins *et al.* (2008) propose to support learners with personalized competence-based recommendations. The authors investigate the efficacy of using competence descriptions in personalized recommender systems. For a systematic review of competence-based recommender systems, refer to Yago *et al.* (2018).

Modeling competences also plays an important role in the development of educational recommender systems (Pavlik and Anderson, 2008). An example is given by Mozer and Lindsey (2016), who follow a hybrid approach that integrates collaborative filtering and computational models of forgetting, such as a variant of the above described ACT-R activation equation. More specifically, they use collaborative filtering to infer a student's latent traits, such as the memory strength for a given item (e.g., vocabulary) or the individual time-based memory decay rate. They then exploit the activation equation to predict the student's knowledge state with respect to the item.

Thaker *et al.* (2018) present an approach to model dynamic student knowledge for online adaptive textbooks. Their model integrates student activities in a knowledge tracing framework (Corbett and Anderson, 1994), a framework based on ACT-R to model changes in knowledge states during acquisition of skills. In the work of Thaker *et al.*, students' current level of knowledge is derived from behavioral data and quiz activities.

A mathematically complementary approach can be found in educational recommender systems, which draw on the set-theoretical framework of Knowledge Space

Theory (e.g., Falmagne *et al.* (2013)). Based on the observed problem solving behavior of a student, e.g., in the domain of mathematics, the probability distribution over the underlying subset of knowledge states (i.e., problems that can already be mastered) is estimated. These estimates serve as input for adaptive recommendations of learning objects, which are neither too easy nor too difficult.

2.1.6 Discussion

Incorporating cognitive models of human cognition to design and improve recommender systems is a promising research direction. In particular, a variety of human memory models have been applied to model user behavior and to improve recommender systems. The use of parts of the cognitive architecture ACT-R has put forth effective recommendation systems. The most compelling reason here is that the BLL equation formalizes fundamental time-based memory decay processes in a computationally efficient manner; additionally, its underlying psychological model is intuitive and contributes to a deeper understanding of user behavior. However, recommender systems based on the BLL equation foster interaction with content similar to what a user has already interacted with recently and frequently in the past (e.g., scientific content like in the work of Maanen and Marewski (2009)). Depending on the use case (e.g., recommending political news) this may lead to confirmation bias, i.e., the tendency to recall information that mostly confirms one's existing beliefs. Understanding the implications of such recommender systems from both the user and the system perspective is an open challenge for future research. One strand of research can look into the diversification of recommendation results to mitigate confirmation bias. For an overview of diversification in recommender systems, please refer to Castells *et al.* (2015). The topic of *counterfactual reasoning* (Hoch, 1985) can be another strand of research to alleviate confirmation, and in a larger context, information bias. Counterfactual reasoning is a core concept in human cognition that corresponds to thinking about a past situation and reflecting on alternative outcomes that might also have been. Galinsky and Moskowitz (2000) show in a psychological study that counterfactual reasoning can make study participants explore alternative explanations in situations in which they typically seek confirmatory information. Future work can explore how to develop counterfactual recommendations that help users explore alternative choices and their impact on user behavior.

CBR recommender systems, while being a category of recommender systems on their own, are also built on principles of cognition. In essence, they mimic how humans draw on previous learning episodes when solving new problems. One of their advantages is that they help generate recommendations in a transparent and explainable fashion. However, they require a knowledge base, whose creation is often labor intensive. More research is needed to devise efficient techniques to create and maintain such knowledge bases.

Furthermore, we reviewed works that incorporate a user's attention into the recommendation model. While the success of deep learning has spawned a range of attention-based approaches, we are not aware of any works that discuss underlying psychological models and theories of attention. Here, we see potential for future work to investigate attention-based approaches in light of underlying psychological constructs. That can foster the transparency and interpretability of the inner workings

of such algorithms.

Finally, there is also untapped potential in the study of the connection between utilizing human memory processes to design and improve recommender systems and using recommender systems to support human memory in retrieving objects. While both strands of research agree on the relevance of context cues for determining the importance of objects in human memory, to date, research that addresses both aspects simultaneously is scarce.

2.2 Personality-aware Recommender Systems

Personality is a fundamental human characteristic, which has been studied in psychology for decades. Personality traits are human characteristics that are stable over the years. In contrast to mood or emotion, which change frequently and are context-dependent, personality traits do not depend on a particular context or stimulus. Personality traits are known to be significantly correlated with user characteristics that recommender systems exploit, such as music preferences (Tkalcic and Chen, 2015a; Ferwerda *et al.*, 2017b), or preferences for movies (Golbeck and Norris, 2013) or books (Rentfrow *et al.*, 2011); or the need for diversity in recommendation lists (Chen *et al.*, 2013b; Wu *et al.*, 2013). Nguyen *et al.* (Nguyen *et al.*, 2018) find, in a user study with over 1,800 subjects, that personality traits of users can also help infer the users' preferences for recommendation diversity, popularity, and serendipity. They also show that user satisfaction increases when personality traits are incorporated into the recommendation process. The correlation between user preferences and personality traits is also confirmed in the work of Karumur *et al.* (Karumur *et al.*, 2018). In a user study conducted on the MovieLens dataset,¹ the authors identify user behavior that is related to the recommender system (i.e., user retention and engagement, preferences, and rating patterns), and show that the personality traits of the users correlate significantly with their behavior.

The most common motivations for considering personality in the recommendation process are to alleviate cold-start situations (in particular for new users) and to improve the level of personalization (e.g., to increase recommendation list diversity). Karumur *et al.* (Karumur *et al.*, 2016) go beyond this research as their aim is to identify specific areas where personality is most likely to provide value in recommender systems. To that end, they study category-by-category variations in preference (both rating levels and distribution) across different personality types.

2.2.1 Eliciting Personality Traits

While a variety of models exist to describe human personality traits, the least disputed and most commonly used model in the context of recommender systems research is the *Five Factor Model* (FFM), which is also known as the *Big Five* model or the *OCEAN* model (McCrae and John, 1992). Please note that other personality models are the Thomas-Kilman conflict mode personality model (Thomas, 1992), which is used to model dynamics in groups (Felfernig *et al.*, 2018d), or the vocational RIASEC model (short for Realistic, Investigative, Artistic, Social, Enterprising,

¹<https://grouplens.org/datasets/movielens>

and Conventional) model (Holland, 1997), which is used to deliver personalized recommendations in an e-commerce setting (Bologna *et al.*, 2013), as well as the Bartle model (short for Killers, Achievers, Explorers, and Socializers) (Stewart, 2011), which is used to provide recommendations in gamified learning settings (Konert *et al.*, 2013; Paiva *et al.*, 2015).

The most commonly adopted FFM describes personality along the five dimensions of openness to experience (conventional vs. creative thinking), conscientiousness (disorganized vs. organized behavior), extraversion (engagement with the external world), agreeableness (need for social harmony), and neuroticism (emotional stability). Various instruments have been developed to elicit personality traits, with questionnaires being a common choice. Each personality dimension is then described along a given multi-point rating scale, e.g., between 1 and 5. To this end, the responses/ratings to the questions are linearly combined using a fixed combination for each trait.

A comprehensive resource for such instruments is the International Personality Item Pool (IPIP)², which contains a wealth of measures and scales (Goldberg *et al.*, 2006).

The most commonly used instruments to elicit personality traits according to the FFM include the *Ten Item Personality Inventory* (TIPI) (Gosling *et al.*, 2003) and the *Big Five Inventory* (BFI) (John and Srivastava, 1999). The former asks users to fill in 10 questions on a 7-point scale from “disagree strongly” to “agree strongly” (e.g., “I see myself as anxious, easily upset.”). The latter, sometimes also referred to as BFI-44, uses 44 questions. Both linearly combine the answers to result in a final score for each of the OCEAN dimensions.

Please note that several approaches exist, which infer personality traits not based on questionnaires; including extracting personality information from eye tracking data (Berkovsky *et al.*, 2019), communication behavior in Web-based learning Systems (Wu *et al.*, 2019), visual and content features from Instagram pictures (Ferwerda and Tkalcic, 2018), or social media content (Golbeck *et al.*, 2011a; Golbeck *et al.*, 2011b; Golbeck, 2016).

2.2.2 Personality Traits in Recommender Systems

Since personality traits are human characteristics that are stable over years and do not depend on a certain context or stimulus, in contrast to mood or emotion, respectively, they can be used to create personalized recommender systems. The most common motivations for considering personality in the recommendation process include to alleviate *cold-start* situations (in particular for new users) and to improve the level of *personalization* (e.g., to increase recommendation list diversity). In the following, we present a selection of very recent work; for a review of earlier research, please consider (Tkalcic and Chen, 2015b).

Asabere *et al.* propose a recommender system for *conference attendees* that integrates personality traits and social ties of attendees (Asabere *et al.*, 2018). Personality is described using the OCEAN model, social ties using contact duration and frequency of conference attendees (they were equipped with smartphones). User

²<https://ipip.ori.org>

similarity in terms of personality is computed using Pearson's correlation between two persons' OCEAN scores treated as a vector. The similarity between two attendees concerning social ties is computed as a product of their contact frequency and duration. Based on these two kinds of similarity, the authors present a hybrid system that linearly combines the personality and social tie similarities between users. The system alleviates cold start for users with low social tie strength (e.g., users who just arrived at the conference) by resorting to using personality only.

Yang and Huang propose a personality-aware recommender for *computer games* (Yang and Huang, 2019). They predict players' OCEAN traits from their social media posts employing methods of personality recognition from texts, in particular, natural language processing techniques. Games are also assigned personality scores based on the personality of their players and based on results of personality recognition applied to game reviews. The target user is then recommended games that are played by users with a similar personality, an approach that resembles memory-based collaborative filtering where similarities are computed over personality trait vectors rather than rating vectors. Alternatively, the target user is recommended games similar to the games the user interacted with, which resembles content-based filtering where game content is modeled by predicted "personality" of the game.

Adaji et al. present a graph-based approach for recommending *recipes* using personality information of users of an online social networking site for cooking (Adaji et al., 2018). The authors extract OCEAN scores from users' reviews and describe each recipe by the dominant personality trait of its reviewers. A graph is then constructed in which nodes (recipes) are connected by edges indicating that the same user has reviewed them. Recipes are weighted by the number of reviews received; edges are weighted by the number of users who reviewed both recipes the edge connects. The authors propose to alleviate cold start by first creating a recipe subgraph that only contains the recipes whose dominant "personality" matches that of the new user. Recipes are then recommended starting at the node with the highest weight and traversing the graph in decreasing order of edge weight.

Nalmpantis and Tjortjis present a simple method to include personality into a *movie* recommender system (Nalmpantis and Tjortjis, 2017). Based on the OCEAN traits of the target user, the authors compute the Manhattan distance between the user's traits and the traits assigned to each genre in a list of movie genres with personality annotations created by (Cantador et al., 2013). The proposed system, which is based on a nearest neighbor collaborative filtering approach, then predicts ratings as a linear combination between the movie ratings predicted by the CF component and the user's personality-based genre distance to the movie's genre under consideration.

Gelli et al. integrate personality information into an *image* recommender system, framed as the task of predicting interactions of users with images shared on Twitter (Gelli et al., 2017). To this end, the authors propose a context-aware factorization machine that integrates both sparse features (user-item interactions) and dense feature vectors, such as multimedia content descriptors or user side information. As dense features, they include the users' OCEAN scores and visual concept vectors of the images under consideration into the model, to learn a joint representation. Personality scores of the Twitter users are extracted from their shared posts using

the Apply Magic Sauce API.³

Personality information is often considered in recommender systems to tailor the level of *diversification* of recommended items to the user's needs, relying on studies that show that personality is correlated with a preference for diversity, e.g., (Tintarev *et al.*, 2013; Wu *et al.*, 2018; Ferwerda *et al.*, 2017a). For instance, Lu and Tintarev propose a *music* recommendation system that adapts to users' personality factors and their diversity needs on music preferences (Lu and Tintarev, 2018). They rerank results of a collaborative filtering approach by linearly combining the original rank of each item (song) produced by collaborative filtering and the degree of diversity that the item contributes to the recommendation list, integrating personality as a weighting term into the objective function used for reranking. The authors describe users' personality according to the OCEAN model and define diversity as intra-list diversity, i.e., averaged pairwise distance between all items in the recommendation list. These distances are computed on item features, namely music key, genre, and the number of artists. In a pilot study, the authors found these three features to be most correlated to personality traits. For instance, extraversion was correlated with diversity in terms of music key, as well as agreeableness and diversity in the number of artists. Based on such correlations, Lu and Tintarev then map each personality factor to a desired level of diversity and integrate this as a weighting term into the objective function used for reranking.

Wu *et al.* propose an approach for recommending *interest groups* to join for users of an online social network (Wu *et al.*, 2018). The approach tackles cold start and tailoring recommendations to the user's desired level of diversity by integrating personality information into a user-based CF system. The authors elicit OCEAN scores and linearly combine user similarity in terms of item ratings and personality-based user similarity to alleviate cold start. The personality-based similarity is defined as the Euclidean distance between two users' personality scores. Adjusting diversity is achieved by integrating findings of a pilot study in which the authors use OCEAN traits to predict diversity preferences of users of a Chinese social network site. Thereby, diversity of a user is measured as entropy over categories of interest groups (e.g., sports or culture) the user joined on the site. The recommender system then adjusts the level of item diversity in the recommendation list so that it best matches the diversity level desired by the target user (as estimated from his or her personality traits).

Fernandez-Tobias *et al.* propose a personality-aware recommender system, which they evaluate for recommending *books*, *movies*, and *music* (Fernandez-Tobias *et al.*, 2016). Among other contributions (e.g., on active learning and cross-domain recommendation), the authors extend the classical matrix factorization approach commonly used in model-based collaborative filtering by integrating a user latent factor that describes their personality in terms of the five dimensions of the OCEAN model. The proposed personality-based matrix factorization approach can deal with implicit feedback data, i.e., information on user-item interactions beyond explicit ratings, such as clicks, purchases, or the frequency of item consumption.

³<https://applymagicsauce.com>

2.2.3 Personality Traits in Group Recommender Systems

Personality can also be taken into account in group recommendation scenarios to improve the quality of group decisions and increase user satisfaction. In groups of users, especially heterogeneous groups, a conflict situation may arise quickly since group members have different personality traits, which leads to contradictions in terms of the preferences of group members (Recio-Garcia *et al.*, 2009). Thereby, generating group recommendations by solely aggregating group members' preferences, using standard social choice functions (Felfernig *et al.*, 2018a; Masthoff, 2011), might not reflect the overall satisfaction of a group (Quijano-Sanchez *et al.*, 2010).

More recently, some novel methods have been proposed to create group recommendations considering different types of group members' personalities. For instance, Rossi and Cervone (2016) propose a group recommendation approach that considers the *agreeableness* factor. The authors argue that in choosing an item in a group of close friends, agreeableness, being related to *altruistic behavior* (Costa and McCrae, 1995), plays a crucial role. An agreeable person tends to compromise and avoid items that are not in the interest of others. Based on this idea, instead of defining a specific social choice function that considers the agreeableness factor, the proposed solution uses the definition of an individual *utility function* to evaluate the item rating of each group member. The underlying idea of this function is "*the user satisfaction if the recommender system chooses that item for the group*". This function conforms to the model proposed by Charness and Rabin (2002) that maximizes the social welfare and increases the sum of group members' payoffs. The utility function measures how much a group member likes to increase social surplus, caring about helping himself/herself and others with low payoffs. In another work, Delic *et al.* (2017) conduct a user study in the travel destination domain to explore the satisfaction levels of individual group members with the final group decision. The authors find out that group members are highly satisfied with the outcome of group negotiations when the final group decision matches their initial preferences. Besides, they indicate that individual satisfaction is correlated with the Big Five personality traits of group members. The satisfaction with the final group decision is positively correlated with the traits *agreeableness* and *conscientiousness* and negatively correlated with the trait *neuroticism*.

Personality traits of group members can also be exploited in group recommender systems to resolve conflict situations in group decisions. Nguyen *et al.* (2019), Quijano-Sanchez *et al.* (2010), and Recio-Garcia *et al.* (2009) characterize the personality of group members using the *Thomas-Kilmann Conflict Mode Instrument* (TKI) model (Thomas and Kilmann, 1974). This model describes a group member's behavior in conflict situations according to two dimensions: *assertiveness* and *cooperativeness*. These dimensions are the extent to which an individual attempts to satisfy his/her own (assertiveness) and other people's preferences (cooperativeness) (Nguyen *et al.*, 2019). The dimensions can be used to define five personality modes of conflict resolution: (i) *competing* (assertive and uncooperative), (ii) *collaborating* (assertive and cooperative), (iii) *avoiding* (unassertive and uncooperative), (iv) *accommodating* (unassertive and cooperative), and (v) *compromising* (moderately assertive and cooperative) (Nguyen *et al.*, 2019; Recio-Garcia *et al.*, 2009). Although these studies share the common idea of exploiting the personality of group members for conflict resolution, they show different points of view in modeling the dimensions *assertiveness*

and *cooperativeness*. Nguyen *et al.* (2019) model *assertiveness* as the probability that group members propose items to the discussion that are highly related to their preferences. Thereby, the probability increases if a group member is assertive and decreases otherwise. In contrast, the authors model *cooperativeness* with the probability that a group member gives positive and negative evaluations to items proposed by other group members. A group member with a *high cooperativeness* tends to have a higher probability of giving positive feedback and a lower probability of giving negative feedback. Quijano-Sánchez *et al.* (2010) and Recio-Garcia *et al.* (2009) estimate the *assertiveness* and *cooperativeness* of a group member based on the sum of the coefficients of his/her personality modes specified by the TKI model (i.e., *competing*, *collaborating*, *avoiding*, *accommodating* and *compromising*). These two dimensions are combined to estimate a *Conflict Mode Weight* (*CMW*) indicating how selfish or cooperative a group member is. The *CMW* value is in the range of [0..1], where “0” reflects a very cooperative person and “1” reflects a very selfish one. The rating of a group member u for a specific item can then be predicted by considering the difference between $CMW(u)$ and any other user v in the group ($CMW(v)$), e.g., in a simple user-based CF fashion. Recio-Garcia *et al.* (2009) also apply a CF approach to first recommend the best items for each group member (we assume $Best_u$ consists of the best items recommended to a group member u using the CF approach). After that, the preferences of individual group members for each item in $Best_u$ are merged using the *minimization misery procedure* (O’Connor *et al.*, 2001). The general idea of this procedure is to minimize as much as possible the misery within the group. For further details of this recommendation approach, we refer to (Recio-Garcia *et al.*, 2009).

2.2.4 Discussion

As illustrated by the reviewed works, personality has a significant impact on user preferences and behavior. The use of personality traits in personalized recommender systems helps alleviate cold-start problems and bears the potential to improve the level of personalization, also in terms of diversification of recommendation results. However, to date, it is not well understood to which extent personality influences perceived recommendation quality; neither is the variability of this extent between users. For some users and domains, tailoring recommendations to personality traits might be valuable to recommend items that fit their personality; for others, personality could be an irrelevant signal, which could even be perceived as invasive concerning privacy and ethics. Incorporating personality in a privacy-aware fashion is an open issue.

Also, current approaches integrate personality in quite simplistic ways, e.g., by linearly combining a content-based similarity with a personality/user-based similarity metric. Only in a very recent article, Beheshti *et al.* (2020) incorporate personality information as features in a neural embedding framework in the larger context of a so-called cognitive recommender system.

Furthermore, manifold instruments and frameworks exist to elicit personality traits. However, the question of when to use which and what quality can be achieved

is still the subject of more detailed investigation. The same holds for the willingness of users to fill out a questionnaire containing tens or even hundreds of questions.

Finally, how to model the “personality” of an item is still an under researched question. More sophisticated methods to derive personality traits on the item level are required. One recent example in this vein is the approach by Sertkan *et al.* (2019).

2.3 Affect-aware Recommender Systems

Affect plays a crucial role in human life. Human affect is commonly categorized into *mood* and *emotion*. Mood refers to an affective experience of longer duration (minutes to hours) but lower intensity, emotion to an affective response of shorter duration (seconds to minutes) to a particular stimulus. Like personality, both mood and emotion are fundamental human characteristics and have been in the focus of psychological research for a long time. They are known to influence our decision making and preferences (Shiv and Fedorikhin, 1999), for example in the domain of videos (Orellana-Rodriguez *et al.*, 2015), fashion (Piazza *et al.*, 2017), music (Ferwerda *et al.*, 2017b), movies (Golbeck and Norris, 2013), or books (Rentfrow *et al.*, 2011); or the need for diversity in recommendation lists (Chen *et al.*, 2013b; Wu *et al.*, 2013) and reading choices in online news (Mizgajski and Morzy, 2019). In addition, consumption of media items plays a vital role for human mood regulation. In the context of music, mood regulation was even identified as one of the main purposes why people listen to music (Schäfer, 2016).

It has also been shown that humans with different personality traits perceive different emotions when listening to the same piece of music (Schedl *et al.*, 2018). Emotion is a well-explored contextual factor in context-aware recommender systems (e.g., (Zheng, 2013)).

2.3.1 Modeling Affect

Focusing on describing emotions, we can distinguish between *categorical* models and *dimensional* models. The former describes emotions using a predefined vocabulary of basic emotion terms (e.g., happy, sad, angry, or relaxed) or secondary emotions that are reactions to primary ones (e.g., energetic, lonely, confused, or hopeful). Dimensional models, in contrast, describe emotions by assigning them values in a continuous space, which is most commonly spanned by the two dimensions *valence* (V) and *arousal* (A), according to Russell (Russell, 1980). Valence refers to the level of the pleasantness of emotion (positive vs. negative), while arousal refers to the emotion’s intensity (high vs. low). The V/A space is sometimes complemented by a third dimension that describes how much in control of the respective emotion a person is (dominant vs. submissive). This dimension is commonly called *dominance* by Mehrabian (Mehrabian, 1980), *potency*, or *control* according to Fontaine *et al.* (Fontaine *et al.*, 2007). An illustration of the valence–arousal plane with several affective terms mapped to it can be found in Figure 2.2.

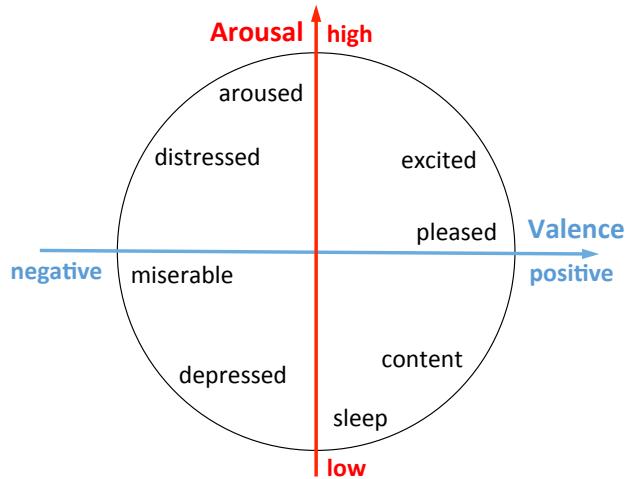


Figure 2.2: Some categorical affective terms mapped to valence–arousal plane (Knees *et al.*, 2019).

2.3.2 Affect in Recommender Systems

To create affect-aware recommender systems, we need to infer the mood or emotion of the user, identify relationships between the user’s affective state and item preferences, and finally match users and items constrained to some function that describes affective relationships (Tkalcic *et al.*, 2011). Most often, both users and items are represented in the same affective space to enable direct computation of similarities between users and items.

Ravi and Vairavasundaram (2017) present a recommender system for *locations* that leverages users’ emotions, and their locations shared in a location-based social network. To establish associations between locations and emotions as well as users and emotions, the authors adopt a lexicon-based approach to identify emotion words in user posts shared at a particular location. Emotions are described using a categorical model of positive emotion categories (happy, like, and surprised) and negative categories (angry, sad, fear, and hate). As a result, each user and each location is described by an emotion vector, which allows computing emotion similarity between users and items. The authors propose adaptations of user-based and item-based collaborative filtering to make recommendations. The user-based collaborative filtering model recommends locations to the target user u based on the product of two components: overall emotional similarity to other users v (irrespective of location) and similarity between u ’s current emotion and the emotion v expressed when visiting the location under consideration. The proposed item-based collaborative filtering model uses the emotionally most similar locations to those locations l already visited by the target user u and weighs them with the similarity between u ’s current emotion and u ’s emotion when visiting l . Cosine similarity is used for all similarity calculations. A hybrid system is also proposed, as a simple linear combination of the user-based

and items-based prediction scores.

Deng *et al.* (2015) propose a similar approach while using other resources and targeting another recommendation domain, i.e., *music*. The authors extract emotion information and music listening information from a popular Chinese microblogging service. They adopt a lexicon-based approach using various Chinese text resources and emoticons to infer emotions of microblogs. Emotions are described using different categorical models of varying granularity (from 2 to 21 emotion categories). To be able to compute similarities, a user's emotional context is then defined by a vector representation over the dimensions of the applied emotion model, where each dimension contains the frequency of terms belonging to the respective emotion category in the user's most recent microblogs. Contextual relationships between emotions and songs for a given user are established by considering the emotions reflected in the user's posts directly before his or her posting of a music listening event. This way, each pair of user and song (listened to by the user) is assigned an emotion vector. For recommending songs, the authors define a user-based collaborative filtering model, an item-based collaborative filtering model, a hybrid of the two, and a random walk model. The former three are almost identical to Ravi and Vairavasundaram (2017), but use songs instead of locations as items. For the random walk approach, the authors construct a bipartite graph of users, emotional contexts (merged by clustering), and songs. They adapt a variant of PageRank to traverse the graph and effect recommendations.

Ayata *et al.* (2018) propose a framework for emotion recognition that can be integrated into *music* recommender systems. The authors gather various physiological signals through wearable sensors (measuring, for instance, skin conductance or heart rate). From the sensor data, several features are computed using different statistical summaries of the physiological measurements (e.g., min, max, mean, variance, median, skewness, and kurtosis) within time windows. These features are used to predict the user's emotional state, where emotions are described using the V/A model. The authors then conceptualize a music recommendation architecture that integrates the affective response of the previous recommended song on the user and adapts future recommendations based on this response.

2.3.3 Discussion

The discussed works show that both emotion and mood are beneficial in context-aware recommendation scenarios, such as location-based recommendations, and in scenarios in which recommended items have a strong affective impact on users, such as music recommender systems. As shown by the literature, users' affective states can be exploited to tailor recommendations to the needs of an individual.

A shortcoming of current research is that it largely neglects dynamic changes in mood or emotion during item consumption. We see further potential to research on detecting such changes and integrating affect dynamics into recommender systems. Besides, as in the case of personality, to which extent a user's mood or emotion influences the perceived recommendation quality is not well understood either and another challenge for future research.

An additional limitation of current work on affect-aware recommenders is that they assign one affective state to the user, neglecting the differences between expressed,

perceived, and induced emotion. This is in contrast to psychological literature, which makes a clear distinction between those kinds of emotions. This distinction is particularly important for recommenders in the entertainment domain, with typical strong emotional attachment of users. Expressed emotion refers to the emotion the creator of an item, such as a photographer or composer, intended to express when creating the item. Perceived emotion refers to the emotion the user (e.g., viewer or listener) perceives when exposed to the item. Induced emotion is the emotion truly experienced or felt by the user. Since these three categories of emotions may be very different (Juslin and Laukka, 2004; Schedl *et al.*, 2018), an emotion-aware recommender system should be able to distinguish between them and incorporate them in multifaceted ways.

Finally, mood and emotion constitute sensitive information. Therefore, more research is needed to make emotion detection and inclusion of emotion as a contextual factor in recommender systems privacy-aware.

3

Recommender Systems and Human Decision Making

So far, in this survey, we focused on recommendation techniques and systems that use psychological features of the user in the recommendation process. In this section, we discuss works that investigate how recommender systems influence human decision making. In addition to helping users make decisions, recommender systems also persuade users (Yoo *et al.*, 2012) and influence human choices. Here, several psychological mechanisms should be taken into account. In the following, we review works that discuss such mechanisms in light of recommender systems research.

When users interact with a recommender system, they make decisions; for instance, they choose an item from the recommendation list (Chen *et al.*, 2013a). Decision making is a fundamental cognitive process that has been studied for decades by renowned psychologists such as Kahneman (Kahneman, 2011), Stanovich (Stanovich and West, 1998), Loewenstein (Loewenstein and Lerner, 2003), Gigerenzer (Gigerenzer and Gaissmaier, 2011), Thaler (Thaler, 1980), or Tversky (Tversky and Kahneman, 1974), who describe the process of users' decision making as being not completely rational (Stanovich and West, 1998; Kahneman, 2003), in cases guided by affect (Loewenstein and Lerner, 2003), influenced by biases and heuristics (Tversky and Kahneman, 1974), and anchor effects (Tversky and Kahneman, 1974), or subject to bounded rationality (Simon, 1966), which means that cognitive limitations of the decision maker impact rational decisions.

Such factors can lead to sub-optimal decision outcomes. The reason for this is that users frequently do not try to optimize a decision outcome, but instead apply decision heuristics (Payne *et al.*, 1993). Bettman *et al.* (1998) describe that while users' preferences evolve in the course of a decision process, they typically cannot state these from the very beginning. Thus, human decision making is more focused on *constructing* preferences than on *eliciting* preferences. Correspondingly, in the case of recommender systems, users often do not know their preferences beforehand but construct and frequently adapt these within the scope of the recommendation process (Mandl *et al.*, 2011). Please note that the Recommender Systems Handbook dedicates a chapter to human decision making and recommender systems (Jameson *et al.*, 2015). Besides, Teppan and Zanker (2015) present an empirical study of several decision biases in recommender systems. They investigate three types of biases, i.e., decoy effects (Teppan and Felfernig, 2009a; Teppan and Felfernig, 2009b), serial position effects (Felfernig *et al.*, 2007), and framing (Tversky and Kahneman, 1981). Adomavicius *et al.* (2013) discuss anchoring effects (Tversky and Kahneman, 1974; Chapman and Johnson, 2002), which influence the decisions of users if they are presented with an initial proposed value for available options.

Furthermore, decision making behavior varies between users. The work of Jameson

et al. (2015) describes a variety of choice patterns observed in users and outlines how recommender systems can support such patterns. Karimi *et al.* (2015) investigate the variance of user decision making behaviors on the basis of analyzing four archetypes of online customers. The authors find that the decision making behavior of users significantly differs depending on the nature of the decisions (i.e., number of cycles, duration, number of alternatives and number of criteria). Jugovac *et al.* (2018) present a study on how to adapt recommender systems to such different decision-making styles.

In the next sections, we summarize research efforts on relevant factors that influence decision making and which can impact the likelihood of recommended items being selected by a user. Besides, we discuss further aspects of counteracting decision biases. Please note that, we focus on approaches to mitigate *decision biases* that occurred in users' interactions with recommender systems. The discussed solutions are user-interface oriented, which help to minimize the impact of decision biases at the rating collection time. In the current literature, there exist various approaches to eliminate biases in datasets, algorithms, and recommendation results; however, they are not our primary focus. For further related details of these approaches, we refer to Chen *et al.* (2020) and Huang *et al.* (2020).

3.1 Decoy Items

One decision bias results from users making decisions depending on the way decision alternatives are presented to them. A frequent decision heuristic in this context is an attribute-wise comparison between items (Payne *et al.*, 1993). For example, the inclusion of items that are entirely inferior to other items in a list of alternatives can trigger changes in choice behaviors. Such inferior items are denoted as *decoy items* (*D*) (Huber *et al.*, 1982), which can be used to increase the selection probability of a *target item* (*T*) and potentially decrease the selection probability of the *competitor item* (*C*). Such an effect is called *context effect* (or decoy effect). An illustration is provided in Figure 3.1. A target item *T* is regarded as a *compromise* between *D* and *C* if it is, for example, significantly less expensive than the decoy item and only has a slightly lower quality. *Asymmetric dominance* is given if the target item dominates the decoy item in all dimensions, whereas the competitor item dominates the decoy item in only one dimension. Finally, an *attraction effect* is triggered if the target item, for example, has a significantly higher quality and is only marginally more expensive.

A detailed analysis of different types of context effects in recommender systems is given in Teppan and Felfernig (2012) and Teppan and Zanker (2015). Here, the authors show that decoy items can be applied to increase the selection share of target items, which raises several ethical issues. Being able to identify decoy items in a result set also enables to de-bias the result set by simply omitting decoy items. Decoy items can also be used to generate explanations in knowledge-based recommendation scenarios, e.g., via an attribute-wise comparison that led to the recommendation of specific items.

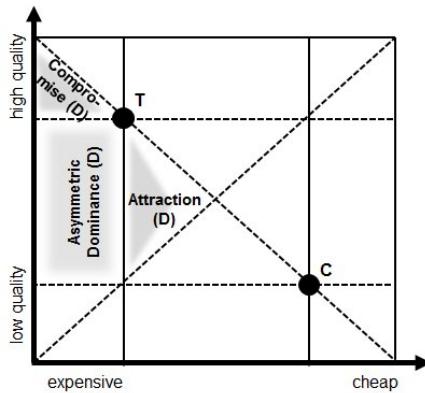


Figure 3.1: An overview of decoy effects. Figure from (Felfernig, 2014).

3.2 Serial Position Effects

Serial position effects can occur in settings where humans are presented with a list of items. These effects have been observed in the context of human memory research (Deese and Kaufman, 1957; Glanzer and Cunitz, 1966) to describe a person's tendency to more likely remember items at the beginning and end of a list (Ranjith, 2012). Figure 3.2 illustrates the effect.

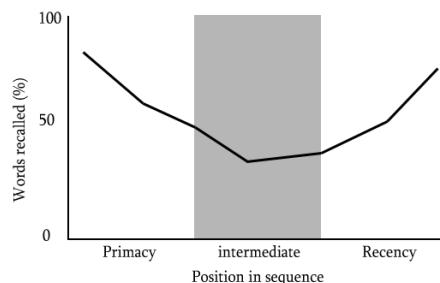


Figure 3.2: This plot (Commons, 2020) shows a U-shaped serial position curve that results from a serial position effect. The effect occurs when a list of words is recalled and words from the start and the end of the list are more likely recalled than words in the middle of the list (Ranjith, 2012).

Murphy *et al.* (2012) show such effects in the context of user link clicking behavior. In their study, links at the beginning of a list were clicked more often than items in the middle of a list, which is called *primacy effect*. Furthermore, there was an increased tendency to click on the links at the end of the list. That is described as the *recency effect*. Primacy and recency effects can also be observed in human memory. Cognitive psychologists showed in memory tests that items at the beginning of a list (primacy) are more easily memorized (Crowder, 2014) since first items having an

advantage over later items because memory capacity is limited (Waugh and Norman, 1965). The last items in a list (recency) are also more easily remembered since they may be still in the short-term memory during the memory test. Felfernig *et al.* (2007) discuss primacy/recency effects in the context of dialogs in knowledge-based recommendation scenarios. They found that product attributes presented to a user at the beginning and the end of a dialog are recalled more often than items in the middle of a list. These attributes are also the preferred criteria when selecting items from a recommendation list. This still holds in situations where unfamiliar product properties are presented at the beginning and the end of a recommendation dialog. Schnabel *et al.* (2016) present a recommendation interface that enables the user to create shortlists of items that the user is currently considering. A user study reveals that the interface helps users memorize and compare choices and that many users explore more instead of being satisfied with the first good item.

Stettinger *et al.* (2015a) analyze the existence of serial position effects in the context of restaurant reviews. The authors show that the same arguments arranged in different orders can lead to significantly different perceptions of restaurant attractiveness. Similar to decoy effects, serial position effects can be used to influence the selection behavior of users. Serial position effects are also investigated in the context of group decision making. Tran *et al.* (2018) investigate serial position effects in scenarios where the same group of users has to solve a series of decision making tasks in different item domains (*low-* and *high-involvement* item domains). The authors examine whether the order of decision tasks result in different decision making behaviors of group members. Related empirical results show that the group recommendation strategy applied in decisions with high related decision efforts tends to be re-used by group members in the follow-up decision with low related decision effort.

Hofmann *et al.* (2014) explore position bias (Joachims *et al.*, 2017) in light of click-based recommender systems evaluation. Position bias is a problem in click-based evaluation, since the probability that an item will be clicked is influenced by its relevancy and its position in the recommendation list. Related work finds that the probability that an item of a top-N list is clicked decays with its rank (Craswell *et al.*, 2008). Hofmann *et al.* (2014) find that if no position bias is present, user behavior (i.e., items a user will click) can be predicted based on historic rating data and using error-based metrics such as precision. However, if position biases exist, the performance of the recommender systems can be wrongly estimated if a performance metric is chosen that does not well reflect the actual user behavior.

In order to counteract serial position effects in group decision making, Stettinger *et al.* (2015b) proposed a solution that allows group members to evaluate items based on MAUT (*Multi-Attribute Utility Theory*) dimensions (Dyer, 2005). With this approach, a group member evaluates an item by articulating his/her preferences for different dimensions that describe the item. The authors conducted a user study in the restaurant domain, where a restaurant was evaluated based on the following dimensions: “ambience”, “price”, “quality”, and “location”. A MAUT-based group recommendation for a specific item is the sum of *individual MAUT values* of group members in the group decision task. The individual MAUT value of a group member is a weighted average of all personal ratings of an item’s dimensions. In the user study, participants were shown a list of restaurants. Each restaurant was described

by a list of arguments describing the restaurants. The arguments were tailored in two types: (1) the *negative salient description* where the negative arguments of the restaurant were placed at the beginning and the end of the description, and (2) the *positive salient description* where the positive arguments of the restaurant were placed at the beginning and the end of the description. The participants were asked to evaluate the restaurant based on the aforementioned dimensions. The user study aimed to examine if the participants' item evaluations were different according to the description type. The experimental results show no significant differences in terms of evaluation values between the two description types. In other words, adopting the MAUT strategy in the item evaluation phase can help to counteract position effects in group decision making.

3.3 Framing Effects

Framing corresponds to the principle that human decisions are influenced by the way options are presented through different wordings, settings, and situations (Tversky and Kahneman, 1981). Framing often comes in the form of gains or losses, as in prospect theory (Tversky and Kahneman, 1992). This theory demonstrates that a loss is perceived as more significant and more worthy of avoiding than an equivalent gain. In the hierarchy of choice architecture, a sure gain is preferred to a probable one, and a probable loss is preferred to a sure loss. Choices can also be worded in a way that highlights the positive or negative aspects of the same decision, and thus prompting affective user responses. The paper by Mandl *et al.* (2011) gives a concise overview of the use of different types of framing effects in recommender systems.

3.4 Anchoring Effects

Anchoring effects are a cognitive bias that makes users rely on the first piece of information (i.e., the anchor) they receive when making subsequent decisions. As pointed out in different social psychology studies, early preference visibility can harm the quality of a decision outcome (Mojzisch and Schulz-Hardt, 2010). Adomavicius *et al.* (2011) find evidence for the anchoring effect in a collaborative filtering scenario. Specifically, they show that anchoring effects can be triggered by disclosing the average rating of similar users. This is verified in a user study presented by Zhang (2011). Köcher *et al.* (2019) provide evidence for a so-called attribute-level anchoring effect that can bias the choices of users towards numerical attributes of product recommendations. Adomavicius *et al.* (2014) also present an approach to de-bias ratings to mitigate anchoring effects using a post-hoc algorithm, as well as a user interface to minimize anchoring biases already when ratings are collected.

Anchoring effects can also be triggered in group recommendation scenarios when one group member's evaluations for items are influenced by the evaluations articulated earlier by other group members. Social-psychological studies point out the correlation between anchoring biases with *confirmation biases*, in which group members tend to focus on discussing available information rather than exploring and sharing new decision-relevant information (Felfernig *et al.*, 2018b). To investigate the impact of anchoring effects in group decision making, Stettinger *et al.* (2015a) conducted

a user study in the requirements engineering where groups of stakeholders had to decide on which requirements should be implemented in their software project. The authors showed that the occurrence probability of an anchoring effect increases if individual group members' preferences are disclosed to others in the early phase of the group decision making process. This brings the idea of counteracting anchoring effects that the preference disclosure should be performed after group members have articulated their preferences for items. The authors also proved that a late preference disclosure helps to increase the group decision performance in terms of user satisfaction, the perceived degree of decision support, the understandability of group recommendations, and the consideration of individual group members' preferences.

3.5 Nudging

Nudging is a concept from behavioral economics to influence human behavior via suggestions towards choices in the users' and societies' long-term interests (Thaler and Sunstein, 2009). *Nudges* are interventions that aim to predictably influence human behavior without limiting any options or significantly changing people's economic incentives (Thaler *et al.*, 2013). Several psychological effects are exploited in nudging that impact decision making, such as those discussed in this chapter, including decision heuristics, the anchoring effect, decoy effects, framing, or the availability and similarity heuristics (Tversky and Kahneman, 1974). Given such effects, a choice architecture, i.e., an environment in which people make decisions (Thaler and Sunstein, 2009), is designed that guides people to decisions that are to their and society's advantage (Jesse and Jannach, 2021). Various such effects are described, and the survey paper by Jesse and Jannach (2021) gives a comprehensive overview of the underlying psychological phenomena of nudging.

Recommendations can be seen as a form of nudging, where the aim is to recommend items that support the nudging goal (Karlsen and Andersen, 2019). Please note that the paper by Jesse and Jannach (2021) gives a structured review of the state-of-the-art of nudging with recommender systems, for a review of nudging in human-computer interaction research, see Caraban *et al.* (2019).

Karlsen and Andersen (2019) present an architecture for nudging recommender systems. As an illustrative example, the authors define the use of nudges to convince people to use environmentally friendly transportation. They introduce a nudge-driven filtering technique that recommends activities to a nudging goal (e.g., use environmentally friendly transportation). The activity is recommended based on a user profile that contains user characteristics and their history of previous activities and behaviors, the user's current context, and the next planned activities. To present the nudges to the user, the authors exploit several decision biases, i.e., framing, anchors, reminders, or social norms (e.g., showing how many others have chosen a particular option (Starke *et al.*, 2020)). Elsweiler *et al.* (2017) aim to nudge people to make better health decisions by recommending them healthy content. In this work, the authors investigate if food recommender systems can nudge users of an online recipe platform towards selecting healthier meals. First, they study if meals in the platform can be replaced with similar but healthier options (in terms of fat content) that also receive high ratings. Then, they conduct a user study to identify if users can distinguish between unhealthy and healthy dishes and find that many cannot tell the

difference, still users tend to select the unhealthier option. In addition, they examine how cues such as recipe title, an image of the meal, and a list of ingredients influence users when selecting recipes and show that users can be nudged to choose healthy over unhealthy recipes; this works particularly well based on visual cues. Esposito *et al.* (2017) introduce nudges to prevent customers in a digital marketplace to purchase incompatible products. In a user study, they evaluate different types of nudges and find that nudges in the form of emotive warning messages and incompatibility information at the checkout page reduce the number of incompatibility purchases. Turland *et al.* (2015) aim to nudge users towards selecting more secure public wireless networks by recommending more secure network alternatives. They found evidence for a decoy effect that nudged users in choosing a secure network.

3.6 Discussion

As pointed out by Teppan and Zanker (Teppan and Zanker, 2015), current recommender systems typically cannot control decision biases, and more research is needed in this direction. From a user perspective, the awareness of the existence of decision biases is essential to make more informed decisions when interacting with a recommender system. Psychology-informed recommender systems that are aware of decision biases can help educate users and make them aware of their own biases in decision making, e.g., via explanations. Another possibility for future research is to study how biases change over time and how these changes impact a user's preferences and behavior.

System-induced biases such as popularity biases (Cremonesi *et al.*, 2010) can be reinforced when already popular items are always put on top of a recommendation list (i.e., exploiting serial position effects). Future research can investigate whether different user groups experience different recommendation utility due to biases such as popularity and demographic biases (Ekstrand *et al.*, 2018). Also, while most related work has been on detecting decision biases in recommendation scenarios, we need more research on proactively preventing or minimizing such biases.

Furthermore, commercial recommendation platforms often actively exploit decision biases of humans to nudge users to adopt a specific behavior or to persuade users to make particular decisions, e.g., what to buy or what to read. This can be beneficial to the user, e.g., when a relevant recommended item is presented prominently. However, it can also be harmful, e.g., in case of decision manipulation (Tran *et al.*, 2019b) and since not all nudges and persuasive mechanisms are helpful and to the user's advantage. For example, marketers may employ nudges to guide consumers towards non-essential options (Schneider *et al.*, 2018). Consequently, ethical concerns and discussions around the concept of nudging and persuasion have emerged (Sunstein, 2015). These discussions gave room for a competing framework termed *boosting* by Grüne-Yanoff and Hertwig (2016). Boosting attempts to help users improve their competencies in decision making instead of nudging them (Hertwig and Grüne-Yanoff, 2017). Grüne-Yanoff *et al.* (2018) distinguishes between boosting and nudging in two aspects: firstly, boosting aims to expand people's competencies by overcoming hu-

man cognitive limitations rather than exploiting them. Secondly, a nudge intervenes in a person's choice environment and exploits specific decision heuristics to guide behavioral change, boosting intervenes on people's decision heuristics and expands their decision competencies to foster a specific behavioral change. While, to the best of our knowledge, boosting has not yet been explicitly employed in recommender systems research, there exist related examples in the information retrieval community - e.g., Zimmerman *et al.* (2020) and Ortloff *et al.* (2021) employ boosting to boost users' competencies in searching while preserving their privacy. Bateman *et al.* (2012) provide a search dashboard to make users reflect on their search behavior by comparing it to the behavior of expert searchers. Moraveji *et al.* (2011) boost the search skills of participants in a user study by offering them tips on conducting optimal searches. In a follow-up study, the authors find that the study participants retain their improved search skills compared to a control group also in the absence of search tips.

We believe that boosting is a promising research area for the recommender systems community. For example, boosting can be applied to improve user knowledge about decision biases and underlying mechanisms of the recommender systems, including the implications of users' behavior on the prediction quality. An advantage would be that some of the ethical concerns that come with nudging and persuasion in recommender systems could be alleviated as well.

4

User-centric Recommender Systems Evaluation

This chapter discusses research works that investigate recommender systems' evaluation with a particular focus on the user perspective. In addition, we review factors that influence how users experience and engage with recommender systems. In the next paragraphs, we, nevertheless, briefly summarize core concepts of classic evaluation metrics and strategies.

Recommendation evaluation has traditionally centered on the accuracy of algorithms (Pu *et al.*, 2012) by quantifying the relevance of recommendations to a user's preferences. To that end, typically, metrics of accuracy are employed such as precision, recall, or normalized discounted cumulative gain (Herlocker *et al.*, 2004). Please note that the survey of Gunawardana and Shani (2009) provides a detailed discussion of accuracy metrics in recommender systems research.

Classic recommender systems evaluation employs either offline, online evaluation (i.e., A/B testing), user studies, or a combination of these methods. In offline evaluation, a pre-collected dataset consisting of user-item interactions is leveraged to simulate users' behavior interacting with a recommender system (Shani and Gunawardana, 2011). Online evaluation corresponds to observing user behavior in real-world, deployed systems (Shani and Gunawardana, 2011). User studies denote an evaluation scenario where small groups of users interact with the recommender system and report their experience (Shani and Gunawardana, 2011). Please note that the respective chapter on evaluation in the Recommender Systems Handbook gives a concise overview of both recommendation evaluation metrics and commonly adopted evaluation strategies (Shani and Gunawardana, 2011).

Related work has discussed that accuracy as a sole metric is not sufficient to assess a recommender system's quality as accurate recommendations might not be perceived as the most useful recommendations (McNee *et al.*, 2006a; Herlocker *et al.*, 2004; Konstan and Riedl, 2012). As a remedy, a variety of so-called beyond-accuracy metrics have been introduced to quantify aspects beyond algorithmic performance. These metrics include *diversity* (Ziegler *et al.*, 2005), *coverage* (Herlocker *et al.*, 2017), or *novelty* and *serendipity* (Herlocker *et al.*, 2004). The latter quantifies how interesting, yet unexpected recommendations are for a user (McNee *et al.*, 2006a). Please note that the survey by Kaminskas and Bridge (2016) gives a concise overview of standard beyond-accuracy metrics used in recommender systems research.

4.1 Psychological Aspects of User Experience

Recommender system evaluation from the user perspective requires a systemic approach beyond the investigation of single actors such as algorithms or users and aims to capture the actors' inter-relations and emerging phenomena, such as user experiences (Ekstrand and Willemse, 2016; Knijnenburg *et al.*, 2012a; McNee *et al.*, 2006b). Given that recommender systems' providers aim to motivate users to return to the system, users must build trust and have a positive perception of the system and its outcomes (Chen and Pu, 2005). Hence, the *user experience* with a recommender system has become the subject of research. User experience is defined by Konstan and Riedl (2012) as the delivery of recommender system outputs to users and the interactions of users with recommendations. In studying the user experience, crucial aspects of recommender systems can be unveiled, such as recommender systems' use and perceived value, and factors related to items, users, user-item interactions, which influence the users' decision-making processes (Xiao and Benbasat, 2007). Such factors include users' attitudes and motivations, their perceived trust in the algorithms, and issues related to the perception of recommender systems in general (Shin, 2020).

Related work investigates the user experience of recommender systems in light of various tasks, e.g., to improve preference elicitation (McNee *et al.*, 2003), increase user satisfaction (Ziegler *et al.*, 2005), study user engagement (O'Brien and Toms, 2008), inspire trust in the system (Pu and Chen, 2006), improve recommendation interfaces (Cosley *et al.*, 2003), or quantify how likely a user will return and recommend a novel system (O'Brien and Toms, 2010).

From a psychological perspective, several factors influence how users experience and engage with recommender systems, such as cognitive dissonance (Festinger, 1954), the persuasiveness of the systems (Fogg, 2002; O'keefe, 2015), perceived system qualities related to interaction and interfaces (Pu *et al.*, 2011; Jugovac and Jannach, 2017), or several attitudes and beliefs (Pu *et al.*, 2011). In the following, we discuss these factors in more detail.

4.1.1 Cognitive Dissonance

Cognitive dissonance denotes a cognitive-affective response to being exposed to information that contradicts one's beliefs and values (Festinger, 1954). Users of recommender systems may experience dissonance after reevaluating a choice they made because they followed a recommendation (Surendren and Bhuvaneswari, 2014) or when being confronted with a recommendation inconsistent with their preferences (Schwind *et al.*, 2011). Dissonance is an aversive cognitive-affective state that users attempt to avoid (Surendren and Bhuvaneswari, 2014) and may make them lose trust in the system (Kuan *et al.*, 2007). Schwind *et al.* (2011), however, explore potential benefits of dissonant recommendations. Concretely, they study if recommending dissonant information for controversial issues helps mitigate confirmation bias. In an online user study conducted on Mechanical Turk, they investigate if users select dissonant or consonant recommendations and assess cognitive and affective reactions to these recommendations. In the first experimental condition, the study participants are recommended an argument on a specific topic that is consonant with the participant's view. In the second condition, they receive a recommendation

with an argument that is inconsistent with their belief. The results show that when a consonant argument is recommended, more users select the consonant argument, and a confirmation bias can be observed. When a dissonant argument is recommended, however, users less frequently select the argument. Also, the consonant recommendations receive better evaluations in terms of cognitive and affective states. In later work, Schwind and Buder (2014) show that dissonant recommendations can help de-bias information selection. However, offering dissonant recommendations might also strengthen people's initial beliefs, mainly when the recommendation falls outside the boundaries of what users consider acceptable (Nguyen *et al.*, 2007). Here, future work can investigate the relationship between cognitive dissonance, boosting (see Section 3.6), and *counterfactual thinking* (Roese, 1997). In the case of counterfactual thinking, consumers reflect on how outcomes could have been different if they had made different decisions (Wang *et al.*, 2017).

4.1.2 Persuasion

Persuasion is a communication process in which a person seeks to convince other people to adapt their behavior and attitudes (Fogg, 2002; Perloff, 2020). Persuasion and the earlier described communicative process of nudging (see Section 3.5) are related concepts, which have originated in different communities, persuasion in social psychology (McGuire, 1969), and nudging in economics (Thaler and Sunstein, 2009), and with slightly different aims. While nudging aims to influence a user's behavior in a particular setting, persuasion aims to influence a person's attitude or behavior; for a more detailed comparison of both concepts, please refer to Meske and Potthoff (2017).

Yoo *et al.* (2012) describe a recommendation as being persuasive when it results in a change of the user's behavior or attitude. The authors elaborate that user interactions with a recommender system correspond to a communication process, in which the extent to which a user is influenced depends on four components: (i) the recommender system itself (source), (ii) the recommendation (message), (iii) the user (target), and (iv) the context, in which the recommendation is offered. These components are integral in the communication-persuasion paradigm, and multiple factors in the components impact if a user is persuaded and changes their behavior or attitude. As Gretzel and Fesenmaier (2006) show, persuasion can happen already during preference elicitation since transparent and short elicitation phases positively influence user satisfaction and perceived fit of later recommendations (Jugovac *et al.*, 2018).

Many studies investigate what makes a recommender persuasive. Related work finds the credibility of recommender systems (Yoo and Gretzel, 2011) is a decisive factor in a recommender system's persuasiveness. Nanou *et al.* (2010) observe that the presentation of recommendation lists in the context of movie recommendations influences persuasiveness. They compare top-N recommendation lists with a structured overview of recommendations, in which recommendations are organized by movie genre, and are presented either as purely textual recommendation lists or as multimodal representation of recommendations (text, images, video). The authors measure persuasiveness in terms of users selecting a recommendation. A small-scale user study with 20 users gives evidence that a structured overview of multimodal

recommendations is more persuasive and results in higher user satisfaction in their domain than a textual recommendation list. Cremonesi *et al.* (2012) observe that the perceived novelty of recommendations has higher persuasive power than the perceived accuracy of recommendations (Jugovac *et al.*, 2018). Felfernig *et al.* (2008a) report that the attractiveness of items contributes to a recommender system's persuasiveness and that the use of attraction decoy items can influence a user's decision-making process. Related work also shows that offering explanations to recommendations can make recommendations persuasive (Herlocker *et al.*, 2000; Tintarev and Masthoff, 2012).

From an ethical perspective, persuasive technology raises several questions (Berdichevsky and Neuenschwander, 1999), naturally, also if applied to recommender systems (Milano *et al.*, 2020). Recommender systems providers may offer persuasive recommendations to maximize some business value, which, from a consumer perspective, might be less transparent (Jesse and Jannach, 2021) and hard to resist (Smids, 2012). According to a standard definition of persuasive technology (Fogg, 2002), persuasion is about voluntary change and needs to function without deceiving the user.

4.1.3 Interaction Methods

Several *perceived system qualities related to interaction and interfaces* influence the user experience of recommendations. Knijnenburg and Willemsen (2015) find that the way lists of recommendations are composed and presented to the user strongly impacts user experience. Knijnenburg *et al.* (2011) construct five interaction methods: (i) a top-N recommendation list, (ii) a sort method that lets users sort recommendations by their preferred attribute, (iii) an explicit method that allows users to assign weights to attributes and thus, directly express their preferences, (iv) an implicit method that automatically weights attributes based on the user's browsing history, and (v) a hybrid combination of the explicit and implicit method. In a user study, the authors compare the five interaction methods and assess user interface satisfaction, trust in the system, system effectiveness, understandability, perceived control, and choice satisfaction. They find that most users are most satisfied with a hybrid recommender that combines implicit and explicit preference elicitation, which gives them some control over the system.

Bollen *et al.* (2010) finds that users, when presented a list of recommendations, tend to inspect only the first few items on the list due to the earlier mentioned primacy effect (see Section 3.2). Chen and Pu (2010b) find that presenting recommendations in the form of a grid can mitigate this issue; however, the authors do not discuss the underlying reasons for that. Another work by Chen and Pu (2010a) suggests a category-based interface, in which a user's top-*N* recommendations are shown as the main category, while other categories contain items that help find trade-offs. As shown by Hu and Pu (2011), an interface where recommendations are grouped into categories, which represent trade-off properties among items, can increase perceived recommendation diversity and improve user satisfaction. Ekstrand *et al.* (2014) present a user study in which each user is provided a recommendation list produced by three variants of collaborative filtering (i.e., item-based, user-based, and an SVD-based variant). The users are asked about their perceptions along five dimensions

of interest, i.e., accuracy, personalization, diversity, novelty, and overall satisfaction with the recommendations. Then, they pairwise compare algorithms based on a first-impression preference and a subjective assessment of the recommendation lists for the five dimensions. Also, the users select their preferred algorithm for future use. The authors find that novelty of recommended items negatively influences the perceived usefulness of the recommendations. The diversity of recommendations positively influences if a user chooses a recommendation algorithm. Please note that Jugovac and Jannach (2017) give a detailed overview of relevant work on user interaction in recommender systems.

4.1.4 Attitudes and Beliefs

User-centric factors such as *attitudes and beliefs* also influence how users evaluate recommendations. Attitudes correspond to the perceived overall perception of the recommender system in terms of user satisfaction and trust, while beliefs describe the user's perception of the usefulness, ease of use, and control of the system (Cremonesi *et al.*, 2011a; Pu *et al.*, 2011). Swearingen and Sinha (2002) find that showing familiar recommendations can increase users' trust in the system. The familiarity principle (Zajonc, 1968) is a psychological effect that makes users establish positive preferences for items to which they are frequently and consistently exposed. Bollen *et al.* (2010) present a user study to understand users' perception of recommendation set attractiveness, choice difficulty, and satisfaction with the selected recommendation. Participants answer 29 questions on a 7-point scale, and in addition, their clicks are logged. The authors fit a structural equation model (Ullman and Bentler, 2003) to the data to understand the interplay between recommendation set attractiveness, choice difficulty, and satisfaction with the chosen item. Please note that structural equation modeling techniques are statistical methods that enable to study relationships between independent variables and dependent variables. Bollen *et al.* (2010) find that user satisfaction depends on the attractiveness of the recommendation set and on the difficulty of choosing from this set. Attractiveness is high if the items in the set vary. A low choice difficulty positively influences satisfaction. However, if the user is presented with more attractive sets, the choice difficulty becomes higher. Willemsen *et al.* (2016) investigate the influence of diversity of the item set on choice overload that arises when users have to select from many items. They find that small but diverse item sets help reduce choice overload and result in similar user satisfaction as top-N recommendations. Jin *et al.* (2019) examine the influence of user control over contextual factors incorporated in the recommendation process on perceived quality, diversity, effectiveness, and users' cognitive load. In a user study, they test two conditions: either the participants have no control over a music recommendation algorithm, or they can choose a particular context, i.e., mood, weather, and location, to which recommendations should be tailored. The study results show that users perceive the utility of recommendations differently when they can select a context. In particular, tailoring recommendations to a user's mood positively impacts recommendation quality and diversity.

4.2 Designing User Studies for Recommender Systems and Existing Evaluation Frameworks

Many research works in recommender systems employ offline evaluation studies, which retrospectively analyze available datasets for certain model-based predictions. While offline evaluation helps assess the internal validity of the recommendation model, it only allows for speculations about the actual user experience. Thus, offline evaluation needs to be complemented by methods that also enable insights into (i) latent user states (e.g., a user's perception of the system) and (ii) the ecological validity of the evaluation results. Both aspects can be addressed by user studies, which we outline next.

When running user studies, it is essential to consider principles of psychological measurement theory (Allen and Yen, 2001) and its application to construct reliable and valid self-report scales (McCroskey *et al.*, 1984; Yannakakis and Hallam, 2011). Self-reports are tests and measures that require individuals to report on their behavior, beliefs, or attitudes. Self-reports are beneficial to elicit key aspects of user experiences, such as engagement (e.g., O'Brien and Toms, 2008), which has been shown to reliably predict perceived usability and durability, i.e., how likely a user is to return to and recommend a novel system (O'Brien and Toms, 2010). The construction of self-report scales requires representative samples of participants and latent factor analysis. Here, a classic choice is to perform an Exploratory Factor Analysis (EFA) (Goretzko *et al.*, 2019) to group inter-item correlations into distinct dimensions (for a review and guideline of how to apply an EFA in the context of recommender systems, see, e.g., O'Brien and Toms, 2010). Subsequently, a Confirmatory Factor Analysis can be run on an independent dataset (gathered from additional user studies) to validate the previously explored factor structure.

For the systematic planning of user studies of recommender systems, Knijnenburg and Willemsen (2015) present a framework that facilitates the design of user studies. In particular, the framework describes how *objective system aspects* such as the algorithm or a presentation layout are perceived by the user and how the user's *perception* – denoted *subjective system aspects* (e.g., perceived recommendation quality and variety), in combination with *personal* and *situational characteristics*, influence the *user experience* and *interaction* with the recommender system. The situational and personal characteristics help account for context-relevant information (e.g., the user's current information goal) and individual variables (e.g., personality traits; see Section 2.2.2).

Similarly, Pu *et al.* (2011) introduce the *ResQue* framework to assess the perceived recommendation quality (e.g., attractiveness, novelty, diversity, and perceived accuracy of recommended items), usability, interface adequacy (e.g., information sufficiency and layout clarity), interaction quality (e.g., preference elicitation and revision), and the overall user satisfaction with the recommender system, as well as the influence of these aspects on the user's intention to buy recommended products and to revisit the system.

To conclusively explain observed effects such as perceived quality differences between algorithms, behavioral data such as user's interactions with recommendations can be triangulated with self-report data. For example, Knijnenburg *et al.* (2012b) demonstrate that two types of matrix factorization algorithms have the same

effects on certain experience variables (e.g., perceived system effectiveness) but are mediated by different subjective system aspects (e.g., different dynamics between perceived diversity and quality). Such study outcomes are essential for the design and improvement of user interfaces, but can only result from the triangulation of behavioral and self-report data.

Due to its high degree of abstraction, the Knijnenburg et al. framework can help guide the systematic planning of research designs. Also, the framework provides a scheme for the formulation of experimentally testable research questions, and it provides guiding information for implementing and analyzing the planned research design, including the operationalization and measurement of the variables to be investigated (e.g., constructing new or using existing questionnaires) and a comprehensive data analysis (e.g., applying structural equation modeling techniques (Ullman and Bentler, 2003)).

4.3 Discussion

The reviewed works show that many factors influence how users experience and engage with recommender systems. One of them is cognitive dissonance. While dissonant recommendations that are inconsistent with the users' attitude could make them lose trust in the system, Schwind and Buder (2014) find that dissonant recommendations can help de-bias information selection. Future work can take a cognitive-computational perspective on biased information behavior (e.g., inspired by self-directed search (Dubey and Griffiths, 2020)) to design recommender systems that help explore non-confirmatory information. Here, the relationship between experienced novelty and curiosity can be explored (e.g., Dubey and Griffiths (2020)): If novelty is perceived as either very high or rather low, an information seeker's curiosity drops; the optimal level of novelty – the sweet spot on the continuum – arises primarily at a moderate level (see also Berlyne (1966)). Future work can therefore investigate, whether a recommender system, drawing on these cognitive-computational accounts of information behavior, can identify the sweet spot of novelty for a given user-subject combination and help identify resources that make the user curious about and willing to tap into them. Another strand for future research lies in exploring boosting in recommender systems to foster counterfactual thinking to de-bias information selection.

Jannach et al. (2019) state that research in recommender systems should strive towards *impact-oriented algorithms* that address the intended purpose of a system. The impact can be, e.g., to help users make better decisions and to increase user satisfaction. As the authors describe, persuasion can help achieve this goal. However, persuasion in recommender systems can decrease the user's possibility to develop their taste since humans tend to take the default setting if one is offered (Knijnenburg et al., 2016).

In their work, Knijnenburg et al. (2016) call for personalized systems that aim to not just recommend the most relevant items to the user but to help users develop, explore, and understand their personal preferences. To that end, they suggest several recommendation lists that contain only items unrelated to the top-N, which means they are not "good" recommendations (e.g., items that the algorithm predicts a user will dislike or unrated items). Their intuition is that such recommendations will help

the user learn about their taste and preferences. Including unrelated recommendations to increase the user's awareness of their preferences can lead to effective feedback mechanisms for recommender systems. On a more general note, such mechanisms can help users understand what assumptions the algorithm makes about them and enable them to correct such assumptions.

The way recommendations are presented also influences whether users are satisfied with a recommender system and the level of control users have in the process. Studying these questions in the context of psychological theories is still a largely unexplored field and requires more interdisciplinary research efforts.

In this chapter, we also discussed the design of studies for user-centric evaluation. From a methodological perspective, user-centric evaluation entails designing questionnaires and conducting user studies that help uncover intrinsic properties and characteristics of subjective user experiences. User studies are a standard evaluation methodology in psychology that help investigate the impact of system changes under natural conditions and access (latent) user states.

Conducting such studies can be challenging, though. In particular, it can be difficult to gather a sufficiently large sample of participants that allows for drawing significant and meaningful conclusions with a high ecological validity. Ecological validity, in psychology, measures whether we can generalize from behavior observed in an experiment to behavior in real-world settings (Schmuckler, 2001). One issue in recommender systems research is that ecological validity can be low (Sinha, Swearingen, *et al.*, 2001), particularly in field studies, where filling out questionnaires can be time-consuming and a burden for the respondents. The challenge is to design the study so that enough reliable data can be collected and respondents participate, which often requires tending towards simplicity in the user-centric evaluation (Fazeli *et al.*, 2017).

To facilitate the design and conduction of user studies, the research community has introduced several evaluation frameworks, as well as beyond-accuracy metrics that quantify more user-centric aspects of recommender systems, such as novelty, serendipity, or diversity. Optimizing a recommendation system for such metrics can help increase user satisfaction, as in the case of diversification of recommendations.

Furthermore, investigating user experience requires access to a deployed system and users interacting with the system over some time (Konstan and Riedl, 2012). That is particularly challenging as academic research often has limited access to such systems. As a remedy, the research community has put notable efforts to help academics build real-world systems via initiatives such as GroupLens (Resnick *et al.*, 1994), or LensKit (Ekstrand *et al.*, 2011).

5

Conclusion- and Suggestions for Future Research

A substantial amount of research on psychology-informed recommender systems has been conducted in the past years. In this paper, we reviewed such recommender systems along three categories: i.e., *cognition-inspired*, *personality-aware*, and *affect-aware* recommender systems.

As shown by the reviewed works on **cognition-inspired recommender systems**, cognitive models help design and improve recommender systems in various domains. One advantage is that these algorithms are interpretable and transparent. Also, they can give further insights into user behavior grounded in human cognition.

While many works in cognition-inspired recommender systems utilize human memory processes to model and predict user behavior, there is untapped potential in the study of the connection between utilizing human memory processes to design and improve recommender systems and using recommender systems to support human memory in retrieving objects. While both strands of research agree on the relevance of context cues for determining the importance of objects in human memory, to date, research that addresses both aspects simultaneously is scarce.

Furthermore, we reviewed works that incorporate a user's attention into the recommendation model. While the success of deep learning has spawned a range of attention-based approaches, we are not aware of any works that discuss underlying psychological models and theories of attention. Here, we see potential for future work to investigate attention-based approaches in light of underlying psychological constructs.

As illustrated by the reviewed works on **personality-aware recommender systems**, personality has a significant impact on user preferences and behavior. The use of personality traits in personalized recommender systems helps alleviate cold-start problems and can improve the level of personalization and diversification of recommendation results both in single-user and group recommendation scenarios.

However, it is not well understood to which extent personality influences perceived recommendation quality; neither is the variability of this extent between users. For some users and domains, tailoring recommendations to personality traits might be valuable to recommend items that fit their personality; for others, personality could be an irrelevant signal, which could be perceived as invasive concerning privacy and ethics. Incorporating personality in a privacy-aware fashion is an open issue. Also, current approaches integrate personality using quite simplistic ways, e.g., by linearly combining a content-based similarity with a personality/user-based similarity metric. Only in a very recent article, Beheshti *et al.* (2020) incorporate personality information as features in a neural embedding framework in the larger context of a so-called cognitive recommender system. Furthermore, how to model the “personality”

of an item is still an under-researched question. More sophisticated methods to derive personality traits on the item level are required. One related example is the approach by Sertkan *et al.* (2019).

In the context of **affect-aware recommender systems**, our survey shows that incorporating users' affective states can help improve personalization. Both emotion and mood are beneficial in context-aware recommendation scenarios, such as location-based recommendations, and in scenarios in which recommended items have a strong affective impact on users, such as music recommender systems. As in the case of personality, to which extent a user's mood or emotion influences the perceived recommendation quality is, to date, not well understood. Nor is the importance of mood or emotion changes during item consumption. We see further potential to research detecting such changes and integrating affect dynamics into recommender systems. Finally, mood and emotion constitute sensitive information. Therefore, more research is needed to make emotion detection and inclusion of emotion as a contextual factor in recommender systems privacy-aware.

On a more general note, existing methods in personality- and affect-aware recommender systems are relatively simple extensions of standard collaborative filtering or content-based filtering algorithms. We see further potential to study how information about personality, mood, and emotion can be integrated into current state-of-the-art deep learning methods (e.g., (Zhang *et al.*, 2019a; Schedl, 2019)).

Finally, most works discussed in this paper employ standard performance metrics from information retrieval and machine learning for evaluation. Future work can explore what metrics psychology-informed recommender systems can improve beyond accuracy, such as algorithmic fairness or transparency. Here, frameworks like the one presented by Deldjoo *et al.* (2021a) could be applied to evaluate user and item fairness and to devise suitable metrics. More research is also needed on the online performance of psychology-informed recommender systems to better understand whether their recommendations result in higher user satisfaction.

In this paper, we have also discussed the relationship between **human decision making and recommender systems**. A range of decision biases are described in the literature, which influence how users interact with a recommender system. Recommender systems can exploit and strengthen such biases to provide more useful recommendations, or to nudge and persuade users. Such effects require a level of control, in particular when they lead to sub-optimal outcomes. While most related work has focused on detecting decision biases in recommendation scenarios, we need more research on proactively preventing or minimizing such biases.

Also, the ethical concerns and discussions around the concept of nudging and persuasion gave room for *boosting* as a competing framework. Since the aim of boosting is to help users improve their competencies in decision making and overcome human cognitive limitations, we believe that boosting is a promising research area for the recommender systems community. For example, boosting can be applied to improve user knowledge about decision biases and underlying mechanisms of the recommender systems, including the implications of users' behavior on the prediction quality.

In this paper, we also discuss the **user-centric evaluation of recommender systems** and factors that influence how users experience and engage with recommender systems. One of them is cognitive dissonance, which, on the one hand, recommender systems designers should avoid as it can make users lose trust in the

system. On the other hand, it can help de-bias information selection. We see potential for future work to take a cognitive-computational perspective on biased information behavior to design recommender systems that help explore non-confirmatory information. Here, the earlier mentioned boosting could help foster such exploration via dissonant recommendations that spark counterfactual reasoning.

Finally, in this paper, we discussed the design of **user studies for recommender systems evaluation**. Here, psychology has strongly influenced recommender systems research since methodologically, user-centric evaluation employs questionnaires and other instruments to uncover intrinsic properties and characteristics of subjective user experiences.

Conducting such studies with ecological validity in mind can be challenging, in particular, to gather a sufficiently large sample of participants that allows for drawing significant and meaningful conclusions. Here, the community could benefit from increased interdisciplinary cooperation between computer science and psychology to benefit from the rich knowledge in the psychological community on designing user studies that do not overburden users and still result in sufficiently large amounts of data.

To facilitate the design and execution of user studies, the research community has introduced several evaluation frameworks. Nevertheless, such user-centric evaluations require access to real-world systems and the ability to observe long-term user behavior. To mitigate this issue, the research community has put notable efforts to help academics build real-world systems via initiatives such as GroupLens (Resnick *et al.*, 1994).

All in all, even though the past few years have witnessed an increasing awareness of psychological considerations in recommender systems research, we are still far away from considering the recommendation task as a multi-perspective endeavor. While historically, recommender systems research has been tied to business (informatics) and computer science, we argue that it should be similarly intertwined with sociological and psychological research.

Our vision for future recommender systems research is, therefore, to draw from the decent knowledge of these disciplines in the entire workflow of creating and evaluating recommender systems. Corresponding systems should, as a result, holistically consider extrinsic and intrinsic human factors; corresponding research should adopt a genuinely user-centric perspective.

References

- Adaji, I., C. Sharmaine, S. Debrownay, K. Oyibo, and J. Vassileva. 2018. "Personality Based Recipe Recommendation Using Recipe Network Graphs". In: *Social Computing and Social Media. Technologies and Analytics*. Ed. by G. Meiselwitz. Cham: Springer International Publishing, 161–170. ISBN: 978-3-319-91485-5.
- Adomavicius, G. and A. Tuzhilin. 2005. "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions". *IEEE Transactions on Knowledge and Data Engineering*. 17(6): 734–749. ISSN: 1041-4347. doi: [10.1109/TKDE.2005.99](https://doi.org/10.1109/TKDE.2005.99).
- Adomavicius, G., J. Bockstedt, S. P. Curley, and J. Zhang. 2011. "Recommender systems, consumer preferences, and anchoring effects". In: *CEUR Workshop Proceedings*. Vol. 811. Chicago, IL, USA: CEUR-WS. 35–42.
- Adomavicius, G., J. C. Bockstedt, S. P. Curley, and J. Zhang. 2013. "Do Recommender Systems Manipulate Consumer Preferences? A Study of Anchoring Effects". *Information Systems Research*. 24(4): 956–975.
- Adomavicius, G., J. C. Bockstedt, C. Shawn, and J. Zhang. 2014. "De-biasing user preference ratings in recommender systems". In: *Joint Workshop on Interfaces and Human Decision Making for Recommender Systems, IntRS 2014, Co-located with ACM Conference on Recommender Systems, RecSys 2014*. Foster City, CA, USA: CEUR-WS. 2–9.
- Aggarwal, C. C. 2016. "Model-based collaborative filtering". In: *Recommender systems*. Springer. 71–138.
- Aguzzoli, S., P. Avesani, and P. Massa. 2002. "Collaborative case-based recommender systems". In: *European Conference on Case-Based Reasoning*. Berlin, Heidelberg: Springer. 460–474.
- Allen, M. J. and W. M. Yen. 2001. *Introduction to measurement theory*. Waveland Press.
- ALRossais, N. A. and D. Kudenko. 2018. "Evaluating Stereotype and Non-Stereotype Recommender Systems". In: *Knowledge-aware and Conversational Recommender Systems (KaRS) Workshop, co-located with RecSys 2018*. Vancouver, Canada: CEUR-WS. 23–28.
- ALRossais, N. A. 2018. "Integrating Item Based Stereotypes in Recommender Systems". In: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization. UMAP '18*. Singapore, Singapore: ACM. 265–268. ISBN: 978-1-4503-5589-6. doi: [10.1145/3209219.3213593](https://doi.acm.org/10.1145/3209219.3213593). URL: <http://doi.acm.org/10.1145/3209219.3213593>.
- Anderson, J. R. 2005. *Cognitive psychology and its implications (6th edition)*. New York: Worth Publishers.
- Anderson, J. R., M. Matessa, and C. Lebiere. 1997. "ACT-R: A theory of higher level cognition and its relation to visual attention". *Human-Computer Interaction*. 12(4): 439–462.
- Anderson, J. R. 1974. "Retrieval of propositional information from long-term memory". *Cognitive psychology*. 6(4): 451–474.
- Asabere, N. Y., A. Acakpovi, and M. B. Michael. 2018. "Improving Socially-Aware Recommendation Accuracy Through Personality". *IEEE Transactions on Affective Computing*. 9(3): 351–361. ISSN: 1949-3045. doi: [10.1109/TAFFC.2017.2695605](https://doi.org/10.1109/TAFFC.2017.2695605).
- Atas, M., R. Samer, A. Felfernig, T. N. T. Tran, S. P. Erdeniz, and M. Stettinger. 2019. "Socially-Aware Diagnosis for Constraint-Based Recommendation". In: *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*. 121–129.
- Ayata, D., Y. Yaslan, and M. E. Kamasak. 2018. "Emotion Based Music Recommendation System Using Wearable Physiological Sensors". *IEEE Transactions on Consumer Electronics*. 64(2): 196–203. ISSN: 0098-3063. doi: [10.1109/TCE.2018.2844736](https://doi.org/10.1109/TCE.2018.2844736).
- Bateman, S., J. Teevan, and R. W. White. 2012. "The search dashboard: how reflection and comparison impact search behavior". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1785–1794.

- Beel, J. 2015. "Towards effective research-paper recommender systems and user modeling based on mind maps". *PhD thesis*. Otto-von-Guericke Universitaet Magdeburg.
- Beel, J., S. Dinesh, P. Mayr, Z. Carevic, and J. Raghvendra. 2017. *Stereotype and most-popular recommendations in the digital library souipoint*. Humboldt-Universität zu Berlin.
- Beel, J. and S. Langer. 2015. "A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems". In: *International conference on theory and practice of digital libraries*. Poznan, Poland: Springer. 153–168.
- Beel, J., S. Langer, B. Gipp, and A. Nürnberg. 2014. "The architecture and datasets of Docear's Research paper recommender system". *D-Lib Magazine*. 20(11/12): 1–11.
- Beel, J., S. Langer, G. Kapitsaki, C. Breitinger, and B. Gipp. 2015. "Exploring the potential of user modeling based on mind maps". In: *International Conference on User Modeling, Adaptation, and Personalization*. Dublin, Ireland: Springer. 3–17.
- Beheshti, A., S. Yakhchi, S. Mousaeirad, S. M. Ghafari, S. R. Goluguri, and M. A. Edrisi. 2020. "Towards Cognitive Recommender Systems". *Algorithms*. 13(8): 1–27.
- Bellandi, V., P. Ceravolo, F. Frati, J. Maggesi, G. Waldhart, and I. Seerer. 2012. "Design principles for competence-based recommender systems". In: *2012 6th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*. Campione d'Italia, Italy: IEEE. 1–6.
- Berdichevsky, D. and E. Neuenschwander. 1999. "Toward an ethics of persuasive technology". *Communications of the ACM*. 42(5): 51–58.
- Berkovsky, S., R. Taib, I. Koprinska, E. Wang, Y. Zeng, J. Li, and S. Kleitman. 2019. "Detecting personality traits using eye-tracking data". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. Glasgow Scotland UK: ACM. 1–12.
- Berlyne, D. E. 1966. "Curiosity and exploration". *Science*. 153(3731): 25–33.
- Bettman, J., M. Luce, and J. Payne. 1998. "Constructive Consumer Choice Processes". *Journal of Consumer Research*. 25(3): 187–217.
- Blanco-Fernández, Y., M. López-Nores, J. J. Pazos-Arias, and J. García-Duque. 2011. "An improvement for semantics-based recommender systems grounded on attaching temporal information to ontologies and user profiles". *Engineering Applications of Artificial Intelligence*. 24(8): 1385–1397.
- Bollen, D., M. Graus, and M. C. Willemsen. 2012. "Remembering the stars? Effect of time on preference retrieval from memory". In: *Proceedings of the sixth ACM conference on Recommender systems*. Dublin, Ireland: ACM. 217–220.
- Bollen, D., B. P. Knijnenburg, M. C. Willemsen, and M. Graus. 2010. "Understanding choice overload in recommender systems". In: *Proceedings of the fourth ACM conference on Recommender systems*. 63–70.
- Bologna, C., A. C. De Rosa, A. De Vivo, M. Gaeta, G. Sansonetti, and V. Viserta. 2013. "Personality-Based Recommendation in E-Commerce". In: *1st workshop on Emotions and Personality in Personalized Services, co-located with 21st Conference on User Modeling, Adaptation and Personalization*. Rome, Italy: CEUR-WS. 1–6.
- Bousbahri, F. and H. Chorfi. 2015. "MOOC-Rec: a case based recommender system for MOOCs". *Procedia-Social and Behavioral Sciences*. 195: 1813–1822.
- Buckner, C. and J. Garson. 2019. "Connectionism". In: *The Stanford Encyclopedia of Philosophy*. Ed. by E. N. Zalta. Fall 2019. Stanford, US: Metaphysics Research Lab, Stanford University.
- Buder, J. and C. Schwind. 2012. "Learning with personalized recommender systems: A psychological view". *Computers in Human Behavior*. 28(1): 207–216.
- Burke, R. 1999. "The Wasabi Personal Shopper: a case-based recommender system". In: *Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*. USA: American Association for Artificial Intelligence. 844–849.
- Burke, R. 2000a. "A case-based reasoning approach to collaborative filtering". In: *European Workshop on Advances in Case-Based Reasoning*. Trento, Italy: Springer. 370–379.
- Burke, R. 2000b. "Knowledge-based recommender systems". *Encyclopedia of library and information systems*. 69(Supplement 32): 175–186.
- Burke, R. 2002. "Hybrid recommender systems: Survey and experiments". *User modeling and user-adapted interaction*. 12(4): 331–370.
- Burke, R. D., K. J. Hammond, and B. C. Young. 1996. "Knowledge-based navigation of complex information spaces". In: *Proceedings of the national conference on artificial intelligence*. Vol. 462. US: AAAI. 468.
- Cantador, I., I. Fernandez-Tobias, A. Bellogin, M. Kosinski, and D. Stillwell. 2013. "Relating Personality Types with User Preferences Multiple Entertainment Domains". In: *Proceedings of the 1st Workshop on Emotions and Personality in Personalized Services (EMPIRE)*. Rome, Italy: CEUR-WS. 1–16.
- Caraban, A., E. Karapanos, D. Gonçalves, and P. Campos. 2019. "23 ways to nudge: A review of technology-mediated nudging in human-computer interaction". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–15.
- Castells, P., N. J. Hurley, and S. Vargas. 2015. "Novelty and diversity in recommender systems". In: *Recommender systems handbook*. Boston, MA: Springer. 881–918.
- Caverlee, J., L. Liu, and S. Webb. 2010. "The SocialTrust framework for trusted social information management: Architecture and algorithms". *Information Sciences*. 180(1): 95–112.
- Chapman, G. B. and E. J. Johnson. 2002. "Incorporating the irrelevant: Anchors in judgments of belief and value". In: *Heuristics and biases: The psychology of intuitive judgment*. New York: Cambridge University Press. 120–138.
- Charness, G. and M. Rabin. 2002. "Understanding Social Preferences with Simple Test". *The Quarterly Journal of Economics*. 117(3): 817–869. ISSN: 0033-5533.

- Chavarriaga, O., B. Florian-Gaviria, and O. Solarte. 2014. "A recommender system for students based on social knowledge and assessment data of competences". In: *European Conference on Technology Enhanced Learning*. Graz, Austria: Springer. 56–69.
- Chen, J., H. Dong, X. Wang, F. Feng, M. Wang, and X. He. 2020. "Bias and Debias in Recommender System: A Survey and Future Directions". *Transactions on Knowledge and Data Engineering*: 1–20.
- Chen, L., M. De Gemmis, A. Felfernig, P. Lops, F. Ricci, and G. Semeraro. 2013a. "Human decision making and recommender systems". *ACM Transactions on Interactive Intelligent Systems (TiiS)*. 3(3): 17.
- Chen, L. and P. Pu. 2005. "Trust building in recommender agents". In: *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*. Citeseer. 135–145.
- Chen, L. and P. Pu. 2010a. "Experiments on the preference-based organization interface in recommender systems". *ACM Transactions on Computer-Human Interaction (TOCHI)*. 17(1): 1–33.
- Chen, L. and P. Pu. 2010b. "Eye-tracking study of user behavior in recommender interfaces". In: *International Conference on User Modeling, Adaptation, and Personalization*. Springer. 375–380.
- Chen, L., W. Wu, and L. He. 2013b. "How Personality Influences Users' Needs for Recommendation Diversity?" In: *CHI '13 Extended Abstracts on Human Factors in Computing Systems. CHI EA '13*. Paris, France: ACM. 829–834. ISBN: 978-1-4503-1952-2. doi: [10.1145/2468356.2468505](https://doi.acm.org/10.1145/2468356.2468505). URL: <http://doi.acm.org/10.1145/2468356.2468505>.
- Chmiel, A. and E. Schubert. 2018. "Using psychological principles of memory storage and preference to improve music recommendation systems". *Leonardo Music Journal*. 28: 77–81.
- Chong, H.-q., A.-h. Tan, and G.-w. Ng. 2007. "Integrated cognitive architectures: a survey". *Artificial Intelligence Reviews*. 28: 103–130.
- Commons, W. 2020. "File:Serial position.png — Wikimedia Commons, the free media repository". [Online; accessed 7-December-2020]. URL: https://commons.wikimedia.org/w/index.php?title=File:Serial_position.png&oldid=509450814.
- Contreras, D. and M. Salamó. 2020. "A Cognitively Inspired Clustering Approach for Critique-Based Recommenders". *Cognitive Computation*. 12(2): 428–441.
- Corbett, A. T. and J. R. Anderson. 1994. "Knowledge tracing: Modeling the acquisition of procedural knowledge". *User modeling and user-adapted interaction*. 4(4): 253–278.
- Cosley, D., S. K. Lam, I. Albert, J. A. Konstan, and J. Riedl. 2003. "Is seeing believing? How recommender system interfaces affect users' opinions". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 585–592.
- Costa, P. and R. McCrae. 1995. "Primary Traits of Eysenck's P-E-N System: Three- and Five-Factor Solutions". *English (US). Journal of Personality and Social Psychology*. 69(2): 308–317. ISSN: 0022-3514. doi: [10.1037/0022-3514.69.2.308](https://doi.org/10.1037/0022-3514.69.2.308).
- Craswell, N., O. Zoeter, M. Taylor, and B. Ramsey. 2008. "An experimental comparison of click position-bias models". In: *Proceedings of the 2008 international conference on web search and data mining*. 87–94.
- Cremonesi, P., F. Garzotto, S. Negro, A. V. Papadopoulos, and R. Turrin. 2011a. "Looking for "good" recommendations: A comparative evaluation of recommender systems". In: *IFIP Conference on Human-Computer Interaction*. Springer. 152–168.
- Cremonesi, P., F. Garzotto, and R. Turrin. 2012. "Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study". *ACM Transactions on Interactive Intelligent Systems (TiiS)*. 2(2): 1–41.
- Cremonesi, P., Y. Koren, and R. Turrin. 2010. "Performance of recommender algorithms on top-n recommendation tasks". In: *Proceedings of the fourth ACM conference on Recommender systems*. Barcelona, Spain: ACM. 39–46.
- Cremonesi, P., R. Turrin, and F. Airoldi. 2011b. "Hybrid algorithms for recommending new items". In: *Proceedings of the 2nd international workshop on information heterogeneity and fusion in recommender systems*. 33–40.
- Crowder, R. G. 2014. *Principles of learning and memory: Classic edition*. Psychology Press.
- Deese, J. and R. A. Kaufman. 1957. "Serial effects in recall of unorganized and sequentially organized verbal material." *Journal of experimental psychology*. 54(3): 180.
- Deldjoo, Y., V. W. Anelli, H. Zaman, A. Bellogin, and T. Di Noia. 2021a. "A flexible framework for evaluating user and item fairness in recommender systems". *User Modeling and User-Adapted Interaction*: 1–55.
- Deldjoo, Y., T. D. Noia, and F. A. Merra. 2021b. "A survey on adversarial recommender systems: from attack/defense strategies to generative adversarial networks". *ACM Computing Surveys (CSUR)*. 54(2): 1–38.
- Delic, A., J. Neidhardt, L. Rook, H. Werthner, and M. Zanker. 2017. "Researching Individual Satisfaction with Group Decisions in Tourism: Experimental Evidence". In: *Information and Communication Technologies in Tourism 2017*. Ed. by R. Schegg and B. Stangl. Cham: Springer International Publishing. 73–85. ISBN: 978-3-319-51168-9.
- Deng, S., D. Wang, X. Li, and G. Xu. 2015. "Exploring User Emotion in Microblogs for Music Recommendation". *Expert Systems with Applications*. 42(23): 9284–9293. ISSN: 0957-4174. doi: <https://doi.org/10.1016/j.eswa.2015.08.029>. URL: <http://www.sciencedirect.com/science/article/pii/S0957417415005746>.
- Doherty, A. R., K. Pauly-Takacs, N. Caprani, C. Gurrin, C. J. Moulin, N. E. O'Connor, and A. F. Smeaton. 2012. "Experiences of aiding autobiographical memory using the SenseCam". *Human-Computer Interaction*. 27(1-2): 151–174.

- Doyle, D., A. Tsymbal, and P. Cunningham. 2003. "A review of explanation and explanation in case-based reasoning". *Tech. rep.* Trinity College Dublin, Department of Computer Science.
- Dubey, R. and T. L. Griffiths. 2020. "Reconciling novelty and complexity through a rational analysis of curiosity." *Psychological Review*. 127(3): 455.
- Dyer, J. S. 2005. "Mau — Multi Attribute Utility Theory". In: *Multiple Criteria Decision Analysis: State of the Art Surveys*. New York, NY: Springer New York. 265–292. doi: [10.1007/0-387-23081-5_7](https://doi.org/10.1007/0-387-23081-5_7). URL: https://doi.org/10.1007/0-387-23081-5_7.
- Ebbinghaus, H. 1885. *Über das Gedächtnis: Untersuchungen zur Experimentellen Psychologie*. Duncker & Humblot.
- Ebbinghaus, H. 2013. "Memory: A contribution to experimental psychology". *Annals of neurosciences*. 20(4): 155.
- Ekstrand, M. D., F. M. Harper, M. C. Willemse, and J. A. Konstan. 2014. "User perception of differences in recommender algorithms". In: *Proceedings of the 8th ACM Conference on Recommender systems*. 161–168.
- Ekstrand, M. D., M. Ludwig, J. Kolb, and J. T. Riedl. 2011. "LensKit: a modular recommender framework". In: *Proceedings of the fifth ACM conference on Recommender systems*. 349–350.
- Ekstrand, M. D., M. Tian, I. M. Azpiroz, J. D. Ekstrand, O. Anuyah, D. McNeill, and M. S. Pera. 2018. "All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness". In: *Conference on Fairness, Accountability and Transparency*. 172–186.
- Ekstrand, M. D. and M. C. Willemse. 2016. "Behaviorism is not enough: better recommendations through listening to users". In: *Proceedings of the 10th ACM Conference on Recommender Systems*. 221–224.
- Elaine Rich. 1979. "User modeling via stereotypes". *Cognitive Science*. 3(4): 329–354. ISSN: 0364-0213. doi: [https://doi.org/10.1016/S0364-0213\(79\)80012-9](https://doi.org/10.1016/S0364-0213(79)80012-9). URL: <http://www.sciencedirect.com/science/article/pii/S0364021379800129>.
- Elsweiler, D., I. Ruthven, and C. Jones. 2007. "Towards memory supporting personal information management tools". *Journal of the American Society for Information Science and Technology*. 58(7): 924–946.
- Elsweiler, D., C. Trattner, and M. Harvey. 2017. "Exploiting food choice biases for healthier recipe recommendation". In: *Proceedings of the 40th international acm sigir conference on research and development in information retrieval*. 575–584.
- Esposito, G., P. Hernández, R. van Bavel, and J. Vila. 2017. "Nudging to prevent the purchase of incompatible digital products online: An experimental study". *PLoS one*. 12(3): e0173333.
- Falmagne, J.-C., D. Albert, C. Doble, D. Eppstein, and X. Hu. 2013. *Knowledge spaces: Applications in education*. Springer Science & Business Media.
- Farrell, S. and S. Lewandowsky. 2018. *Computational modeling of cognition and behavior*. Cambridge University Press.
- Fazeli, S., H. Drachsler, M. Bitter-Rijpkema, F. Broens, W. Van der Vegt, and P. B. Sloep. 2017. "User-centric evaluation of recommender systems in social learning platforms: accuracy is just the tip of the iceberg". *IEEE Transactions on Learning Technologies*. 11(3): 294–306.
- Fehling, M. R. 1993. "Unified Theories of Cognition: modeling cognitive competence".
- Felfernig, A., M. Atas, D. Helic, T. N. T. Tran, M. Stettinger, and R. Samer. 2018a. "Group Recommender Systems: An Introduction". In: NY, USA: Springer. Chap. Algorithms for Group Recommendation. 27–58.
- Felfernig, A., G. Friedrich, B. Gula, M. Hitz, T. Kruggel, G. Leitner, R. Melcher, D. Riepan, S. Strauss, E. Teppan, and O. Vitouch. 2007. "Persuasive recommendation: Serial position effects in knowledge-based recommender systems". In: *Persuasive Technology*. Vol. 4744. LNCS. Springer. 283–294.
- Felfernig, A. 2014. "Biases in Decision Making." In: *DMRS*. 32–37.
- Felfernig, A., M. Atas, M. Stettinger, T. N. T. Tran, and G. Leitner. 2018b. "Group Recommender Systems: An Introduction". In: NY, USA: Springer. Chap. Biases in Group Decisions. 145–155.
- Felfernig, A., L. Boratto, M. Stettinger, and M. Tkalcic. 2018c. *Group recommender systems: An introduction*. Springer.
- Felfernig, A., L. Boratto, M. Stettinger, and M. Tkalcic. 2018d. "Personality, emotions, and group dynamics". In: *Group Recommender Systems*. Springer. 157–167.
- Felfernig, A. and R. Burke. 2008. "Constraint-based recommender systems: technologies and research issues". In: *Proceedings of the 10th international conference on Electronic commerce*. 1–10.
- Felfernig, A., B. Gula, G. Leitner, M. Maier, R. Melcher, S. Schippel, and E. Teppan. 2008a. "A dominance model for the calculation of decoy products in recommendation environments". In: *AISB 2008 symposium on persuasive technology*. Citeseer. 43–50.
- Felfernig, A., B. Gula, G. Leitner, M. Maier, R. Melcher, and E. Teppan. 2008b. "Persuasion in knowledge-based recommendation". In: *International Conference on Persuasive Technology*. Springer. 71–82.
- Fernandez-Tobias, I., M. Braunhofer, M. Elahi, F. Ricci, and I. Cantador. 2016. "Alleviating the New User Problem in Collaborative Filtering by Exploiting Personality Information". *User Modeling and User-Adapted Interaction*. 26(2-3): 221–255. ISSN: 0924-1868. doi: [10.1007/s11257-016-9172-z](https://doi.org/10.1007/s11257-016-9172-z). URL: [http://dx.doi.org/10.1007/s11257-016-9172-z](https://doi.org/10.1007/s11257-016-9172-z).
- Ferwerda, B., M. P. Graus, A. Vall, M. Tkalcic, and M. Schedl. 2017a. "How Item Discovery Enabled by Diversity Leads to Increased Recommendation List Attractiveness". In: *Proceedings of the Symposium on Applied Computing SAC '17*. Marrakech, Morocco: ACM. 1693–1696. ISBN: 978-1-4503-4486-9. doi: [10.1145/3019612.3019899](https://doi.org/10.1145/3019612.3019899). URL: [http://doi.acm.org/10.1145/3019612.3019899](https://doi.acm.org/10.1145/3019612.3019899).
- Ferwerda, B. and M. Tkalcic. 2018. "Predicting users' personality from instagram pictures: Using visual and/or content features?" In: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. 157–161.

- Ferwerda, B., M. Tkalcic, and M. Schedl. 2017b. "Personality Traits and Music Genres: What Do People Prefer to Listen To?" In: *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization. UMAP '17*. Bratislava, Slovakia: ACM. 285–288. ISBN: 978-1-4503-4635-1. doi: [10.1145/3079628.3079693](https://doi.acm.org/10.1145/3079628.3079693). URL: <http://doi.acm.org/10.1145/3079628.3079693>.
- Festinger, L. 1954. "A theory of social comparison processes". *Human relations*. 7(2): 117–140.
- Fogg, B. J. 2002. "Persuasive technology: using computers to change what we think and do". *Ubiquity*. 2002(December): 2.
- Fontaine, J. R., K. R. Scherer, E. B. Roesch, and P. C. Ellsworth. 2007. "The World of Emotions is not Two-Dimensional". *Psychological Science*. 18(12): 1050–1057. PMID: 18031411. doi: [10.1111/j.1467-9280.2007.02024.x](https://doi.org/10.1111/j.1467-9280.2007.02024.x). URL: <https://doi.org/10.1111/j.1467-9280.2007.02024.x>.
- Fu, W.-T. 2008. "The microstructures of social tagging: a rational model". In: *Proceedings of the 2008 ACM conference on Computer supported cooperative work*. ACM. 229–238.
- Fu, W.-T. and W. Dong. 2012. "Collaborative Indexing and Knowledge Exploration: A Social Learning Model". *IEEE Intelligent Systems*. 27(1): 39–46.
- Fu, W.-T., T. Kannampallil, R. Kang, and J. He. 2010. "Semantic Imitation in Social Tagging". *ACM Trans. Comput.-Hum. Interact.* 17(3): 12:1–12:37. ISSN: 1073-0516. doi: [10.1145/1806923.1806926](https://doi.acm.org/10.1145/1806923.1806926). URL: <http://doi.acm.org/10.1145/1806923.1806926>.
- Fu, W.-T. and T. G. Kannampallil. 2010. "Cognitive Models of User Behavior in Social Information Systems". In: *CHI '10 Extended Abstracts on Human Factors in Computing Systems. CHI EA '10*. Atlanta, Georgia, USA: ACM. 4485–4488. ISBN: 978-1-60558-930-5. doi: [10.1145/1753846.1754180](https://doi.acm.org/10.1145/1753846.1754180). URL: <http://doi.acm.org/10.1145/1753846.1754180>.
- Fum, D., F. D. Missier, and A. Stocco. 2007. "The cognitive modeling of human behavior: Why a model is (sometimes) better than 10,000 words". *Cognitive Systems Research*. 8(3): 135–142. Cognitive Modeling. ISSN: 1389-0417. doi: <https://doi.org/10.1016/j.cogsys.2007.07.001>. URL: <http://www.sciencedirect.com/science/article/pii/S1389041707000265>.
- Galinsky, A. D. and G. B. Moskowitz. 2000. "Counterfactuals as behavioral primes: Priming the simulation heuristic and consideration of alternatives". *Journal of Experimental Social Psychology*. 36(4): 384–409.
- Gelli, F., X. He, T. Chen, and T.-S. Chua. 2017. "How Personality Affects Our Likes: Towards a Better Understanding of Actionable Images". In: *Proceedings of the 25th ACM International Conference on Multimedia. MM '17*. Mountain View, California, USA: Association for Computing Machinery. 1828–1837. ISBN: 9781450349062. doi: [10.1145/3123266.3127909](https://doi.org/10.1145/3123266.3127909). URL: <https://doi.org/10.1145/3123266.3127909>.
- Gemmell, J., G. Bell, R. Lueder, S. Drucker, and C. Wong. 2002. "MyLifeBits: fulfilling the Memex vision". In: *Proceedings of the tenth ACM international conference on Multimedia*. 235–238.
- Gigerenzer, G. and W. Gaissmaier. 2011. "Heuristic decision making". *Annual review of psychology*. 62: 451–482.
- Glanzer, M. and A. R. Cunitz. 1966. "Two storage mechanisms in free recall". *Journal of verbal learning and verbal behavior*. 5(4): 351–360.
- Glushko, R. J., P. P. Maglio, T. Matlock, and L. W. Barsalou. 2008. "Categorization in the Wild". *Trends in Cognitive Sciences*. 12(4): 129–135.
- Golbeck, J. and E. Norris. 2013. "Personality, Movie Preferences, and Recommendations". In: *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ASONAM '13*. Niagara, Ontario, Canada: ACM. 1414–1415. ISBN: 978-1-4503-2240-9. doi: [10.1145/2492517.2492572](https://doi.acm.org/10.1145/2492517.2492572). URL: <http://doi.acm.org/10.1145/2492517.2492572>.
- Golbeck, J., C. Robles, M. Edmondson, and K. Turner. 2011a. "Predicting personality from twitter". In: *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*. IEEE. 149–156.
- Golbeck, J., C. Robles, and K. Turner. 2011b. "Predicting personality with social media". In: *CHI'11 extended abstracts on human factors in computing systems*. 253–262.
- Golbeck, J. A. 2016. "Predicting personality from social media text". *AIS Transactions on Replication Research*. 2(1): 2.
- Goldberg, L. R., J. A. Johnson, H. W. Eber, R. Hogan, M. C. Ashton, C. R. Cloninger, and H. G. Gough. 2006. "The International Personality Item Pool and the Future of Public-domain Personality Measures". *Journal of Research in Personality*. 40(1): 84–96. Proceedings of the 2005 Meeting of the Association of Research in Personality. ISSN: 0092-6566. doi: <https://doi.org/10.1016/j.jrp.2005.08.007>. URL: <http://www.sciencedirect.com/science/article/pii/S0092656605000553>.
- Gong, S. 2009. "Joining case-based reasoning and item-based collaborative filtering in recommender systems". In: *2009 Second International Symposium on Electronic Commerce and Security*. Vol. 1. IEEE. 40–42.
- González, G., B. López, and J. L. De La Rosa. 2002. "The emotional factor: An innovative approach to user modelling for recommender systems". In: *Workshop on Recommendation and Personalization in e-Commerce*. 90–99.
- González, G., B. López, and J. L. De La Rosa. 2004. "Managing Emotions in Smart User Models for Recommender Systems". In: *ICEIS (5)*. 187–194.
- Goretzko, D., T. T. H. Pham, and M. Büchner. 2019. "Exploratory factor analysis: Current use, methodological developments and recommendations for good practice". *Current Psychology*: 1–12.
- Gosling, S. D., P. J. Rentfrow, and W. B. S. Jr. 2003. "A very brief measure of the Big-Five personality domains". *Journal of Research in Personality*. 37(6): 504–528.
- Graus, M. and B. Ferwerda. 2019. "Theory-grounded user modeling for personalized HCI". *Personalized human-computer interaction*.

- Gretzel, U. and D. Fesenmaier. 2006. "Persuasion in Recommender Systems". *Int. J. Electron. Commerce*. 11(2): 81–100. issn: 1086-4415. doi: [10.2753/JEC1086-4415110204](https://doi.org/10.2753/JEC1086-4415110204). URL: <http://dx.doi.org/10.2753/JEC1086-4415110204>.
- Grüne-Yanoff, T. and R. Hertwig. 2016. "Nudge versus boost: How coherent are policy and theory?" *Minds and Machines*. 26(1): 149–183.
- Grüne-Yanoff, T., C. Marchionni, and M. A. Feufel. 2018. "Toward a framework for selecting behavioural policies: How to choose between boosts and nudges". *Economics and Philosophy*. 34(2): 243–266. doi: [10.1017/S0266267118000032](https://doi.org/10.1017/S0266267118000032).
- Güell, M., M. Salamó, D. Contreras, and L. Boratto. 2020. "Integrating a cognitive assistant within a critique-based recommender system". *Cognitive Systems Research*. issn: 1389-0417. doi: <https://doi.org/10.1016/j.cogsys.2020.07.003>. URL: <http://www.sciencedirect.com/science/article/pii/S1389041720300449>.
- Gunawardana, A. and G. Shani. 2009. "A survey of accuracy evaluation metrics of recommendation tasks." *Journal of Machine Learning Research*. 10(12).
- Hamilton, D. L. 1979. "A Cognitive-Attributional Analysis of Stereotyping". In: *Advances in experimental social psychology*. Vol. 12. Elsevier. 53–84.
- Hamilton, D. L. 2015. *Cognitive processes in stereotyping and intergroup behavior*. Psychology Press.
- Hammond, K. J. 2012. *Case-based planning: Viewing planning as a memory task*. Elsevier.
- Harvey, M., M. Langheinrich, and G. Ward. 2016. "Remembering through lifelogging: A survey of human memory augmentation". *Pervasive and Mobile Computing*. 27: 14–26.
- Herlocker, J. L., J. A. Konstan, A. Borchers, and J. Riedl. 2017. "An algorithmic framework for performing collaborative filtering". In: *ACM SIGIR Forum*. Vol. 51. No. 2. ACM New York, NY, USA. 227–234.
- Herlocker, J. L., J. A. Konstan, and J. Riedl. 2000. "Explaining collaborative filtering recommendations". In: *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. 241–250.
- Herlocker, J. L., J. A. Konstan, L. G. Terveen, and J. T. Riedl. 2004. "Evaluating collaborative filtering recommender systems". *ACM Transactions on Information Systems (TOIS)*. 22(1): 5–53.
- Hertwig, R. and T. Grüne-Yanoff. 2017. "Nudging and boosting: Steering or empowering good decisions". *Perspectives on Psychological Science*. 12(6): 973–986.
- Hoch, S. J. 1985. "Counterfactual reasoning and accuracy in predicting personal events." *Journal of Experimental Psychology: Learning, Memory, and Cognition*. 11(4): 719.
- Hofmann, K., A. Schuth, A. Bellogín, and M. De Rijke. 2014. "Effects of position bias on click-based recommender evaluation". In: *European Conference on Information Retrieval*. Springer. 624–630.
- Holland, J. L. 1997. *Making vocational choices: A theory of vocational personalities and work environments*. Psychological Assessment Resources.
- Hu, R. and P. Pu. 2011. "Enhancing recommendation diversity with organization interfaces". In: *Proceedings of the 16th international conference on Intelligent user interfaces*. 347–350.
- Huang, J., H. Oosterhuis, M. de Rijke, and H. van Hoof. 2020. "Keeping Dataset Biases out of the Simulation: A Debiasized Simulator for Reinforcement Learning Based Recommender Systems". In: *Fourteenth ACM Conference on Recommender Systems, RecSys '20*. Virtual Event, Brazil: Association for Computing Machinery. 190–199. ISBN: 9781450375832. doi: [10.1145/3383313.3412252](https://doi.org/10.1145/3383313.3412252). URL: <https://doi.org/10.1145/3383313.3412252>.
- Huber, J., J. W. Payne, and C. Puto. 1982. "Adding asymmetrically dominated alternatives: Violations of regularity and the similarity hypothesis". *Journal of consumer research*. 9(1): 90–98.
- Ingwersen, P. 1984. "Psychological aspects of information retrieval". *Social Science Information Studies*. 4(2-3): 83–95.
- Jameson, A., M. C. Willemsen, A. Felfernig, M. de Gemmis, P. Lops, G. Semeraro, and L. Chen. 2015. "Human decision making and recommender systems". In: *Recommender Systems Handbook*. Springer. 611–648.
- Jannach, D., A. Manzoor, W. Cai, and L. Chen. 2020. "A Survey on Conversational Recommender Systems". *arXiv preprint arXiv:2004.00646*.
- Jannach, D., O. S. Shalom, and J. A. Konstan. 2019. "Towards More Impactful Recommender Systems Research." In: *ImpactRS@RecSys*.
- Jannach, D., M. Zanker, A. Felfernig, and G. Friedrich. 2010. *Recommender systems: an introduction*. Cambridge University Press.
- Jannach, D., M. Zanker, M. Ge, and M. Gröning. 2012. "Recommender systems in computer science and information systems—a landscape of research". In: *International Conference on Electronic Commerce and Web Technologies*. Springer. 76–87.
- Jesse, M. and D. Jannach. 2021. "Digital nudging with recommender systems: Survey and future directions". *Computers in Human Behavior Reports*. 3: 100052.
- Jin, Y., N. N. Htun, N. Tintarev, and K. Verbert. 2019. "ContextPlay: Evaluating User Control for Context-Aware Music Recommendation". In: *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2019, Larnaca, Cyprus, June 9-12, 2019*. Ed. by G. A. Papadopoulos, G. Samaras, S. Weibelzahl, D. Jannach, and O. C. Santos. ACM. 294–302. doi: [10.1145/3320435.3320445](https://doi.org/10.1145/3320435.3320445). URL: <https://doi.org/10.1145/3320435.3320445>.
- Joachims, T., L. Granka, B. Pan, H. Hembrooke, and G. Gay. 2017. "Accurately interpreting clickthrough data as implicit feedback". In: *ACM SIGIR Forum*. Vol. 51. No. 1. ACM New York, NY, USA. 4–11.
- John, O. and S. Srivastava. 1999. "The Big Five Trait Taxonomy: History, Measurement, and Theoretical Perspectives". In: *Handbook of Personality: Theory and Research*. Ed. by L. A. Pervin and O. P. John. 2nd. New York, NY, USA: Guilford Press. 102–138.
- Jones, M. N. 2016. *Big data in cognitive science*. Psychology Press.

- Jorro-Aragonese, J., M. Caro-Martinez, J. A. Recio-Garcia, B. Diaz-Agudo, and G. Jimenez-Diaz. 2019. "Personalized Case-Based Explanation of Matrix Factorization Recommendations". In: *International Conference on Case-Based Reasoning*. Springer. 140–154.
- Jugovac, M. and D. Jannach. 2017. "Interacting with recommenders—overview and research directions". *ACM Transactions on Interactive Intelligent Systems (TiiS)*. 7(3): 10.
- Jugovac, M., I. Nunes, and D. Jannach. 2018. "Investigating the decision-making behavior of maximizers and satisficers in the presence of recommendations". In: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. ACM. 279–283.
- Juslin, P. and P. Laukka. 2004. "Expression, perception, and induction of musical emotions: A review and a questionnaire study of everyday listening". *Journal of New Music Research*. 33(2): 217–238.
- Kahana, M. J. 2020. "Computational Models of Memory Search". *Annual Review of Psychology*. 71: 107–138.
- Kahneman, D. 2003. "Maps of bounded rationality: Psychology for behavioral economics". *American economic review*. 93(5): 1449–1475.
- Kahneman, D. 2011. *Thinking, fast and slow*. New York: Farrar, Straus and Giroux.
- Kahneman, D. and A. Tversky. 1972. "Subjective probability: A judgment of representativeness". *Cognitive psychology*. 3(3): 430–454.
- Kaminskas, M. and D. Bridge. 2016. "Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems". *ACM Transactions on Interactive Intelligent Systems (TiiS)*. 7(1): 1–42.
- Karimi, S., K. N. Papamichail, and C. P. Holland. 2015. "The effect of prior knowledge and decision-making style on the online purchase decision-making process: A typology of consumer shopping behaviour". *Decision Support Systems*. 77: 137–147.
- Karlsen, R. and A. Andersen. 2019. "Recommendations with a nudge". *Technologies*. 7(2): 45.
- Karumur, R. P., T. T. Nguyen, and J. A. Konstan. 2016. "Exploring the value of personality in predicting rating behaviors: a study of category preferences on movielens". In: *Proceedings of the 10th ACM conference on recommender systems*. 139–142.
- Karumur, R. P., T. T. Nguyen, and J. A. Konstan. 2018. "Personality, user preferences and behavior in recommender systems". *Information Systems Frontiers*. 20(6): 1241–1265.
- Knees, P., M. Schedl, B. Ferwerda, and A. Laplante. 2019. "User awareness in music recommender systems". *Personalized Human-Computer Interaction*: 223–252.
- Knijnenburg, B. P., N. J. Reijmer, and M. C. Willemsen. 2011. "Each to his own: how different users call for different interaction methods in recommender systems". In: *Proceedings of the fifth ACM conference on Recommender systems*. 141–148.
- Knijnenburg, B. P., S. Sivakumar, and D. Wilkinson. 2016. "Recommender systems for self-actualization". In: *Proceedings of the 10th acm conference on recommender systems*. 11–14.
- Knijnenburg, B. P. and M. C. Willemsen. 2015. "Evaluating recommender systems with user experiments". In: *Recommender Systems Handbook*. Springer. 309–352.
- Knijnenburg, B. P., M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell. 2012a. "Explaining the user experience of recommender systems". *User Modeling and User-Adapted Interaction*. 22(4-5): 441–504.
- Knijnenburg, B. P., M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell. 2012b. "Explaining the user experience of recommender systems". *User Model. User Adapt. Interact.* 22(4-5): 441–504. doi: [10.1007/s11257-011-9118-4](https://doi.org/10.1007/s11257-011-9118-4). URL: <https://doi.org/10.1007/s11257-011-9118-4>.
- Köcher, S., M. Jugovac, D. Jannach, and H. H. Holzmüller. 2019. "New hidden persuaders: an investigation of attribute-level anchoring effects of product recommendations". *Journal of Retailing*. 95(1): 24–41.
- Kolodner, J. 2014. *Case-based reasoning*. Morgan Kaufmann.
- Kolodner, J. L. 1992. "An introduction to case-based reasoning". *Artificial intelligence review*. 6(1): 3–34.
- Konert, J., S. Göbel, and R. Steinmetz. 2013. "Modeling the player, learner and personality: Independency of the models of Bartle, Kolb and NEO-FFI (Big5) and the implications for game based learning". In: *Proceedings of the 7th European Conference on Game Based Learning (ECGBL)*. 329–335.
- Konstan, J. A. and J. Riedl. 2012. "Recommender systems: from algorithms to user experience". *User modeling and user-adapted interaction*. 22(1-2): 101–123.
- Kopeinik, S., D. Kowald, I. Hasani-Mavriqi, and E. Lex. 2017a. "Improving Collaborative Filtering Using a Cognitive Model of Human Category Learning". *The Journal of Web Science*. 4(2): 45–61.
- Kopeinik, S., D. Kowald, and E. Lex. 2016. "Which algorithms suit which learning environments? a comparative study of recommender systems in tel". In: *European Conference on Technology Enhanced Learning*. Springer. 124–138.
- Kopeinik, S., E. Lex, P. Seitlinger, D. Albert, and T. Ley. 2017b. "Supporting collaborative learning with tag recommendations: a real-world study in an inquiry-based classroom project". In: *Proceedings of the Seventh International Learning Analytics & Knowledge Conference*. ACM. 409–418.
- Koren, Y. and R. Bell. 2015. "Advances in collaborative filtering". In: *Recommender systems handbook*. Springer. 77–118.
- Kowald, D., S. Kopeinik, and E. Lex. 2017a. "The TagRec Framework As a Toolkit for the Development of Tag-Based Recommender Systems". In: *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization. UMAP '17*. Bratislava, Slovakia: ACM. 23–28. ISBN: 978-1-4503-5067-9. doi: [10.1145/3099023.3099069](https://doi.acm.org/10.1145/3099023.3099069). URL: [http://doi.acm.org/10.1145/3099023.3099069](https://doi.acm.org/10.1145/3099023.3099069).
- Kowald, D. and E. Lex. 2015. "Evaluating Tag Recommender Algorithms in Real-World Folksonomies: A Comparative Study". In: *Proceedings of the 9th ACM Conference on Recommender Systems. RecSys '15*. Vienna, Austria: ACM. 265–268. ISBN: 978-1-4503-3692-5. doi: [10.1145/2792838.2799664](https://doi.acm.org/10.1145/2792838.2799664). URL: [http://doi.acm.org/10.1145/2792838.2799664](https://doi.acm.org/10.1145/2792838.2799664).

- Kowald, D. and E. Lex. 2016. "The influence of frequency, recency and semantic context on the reuse of tags in social tagging systems". In: *Proceedings of the 27th ACM Conference on Hypertext and Social Media*. ACM. 237–242.
- Kowald, D., E. Lex, and M. Schedl. 2019. "Modeling artist preferences for personalized music recommendations". In: *Proceedings of the Late-Breaking-Results Track of the 20th Annual Conference of the International Society for Music Information Retrieval (ISMIR 2019)*. 1–2.
- Kowald, D., E. Lex, and M. Schedl. 2020a. "Utilizing Human Memory Processes to Model Genre Preferences for Personalized Music Recommendations". In: *Fourth HUMANIZE workshop on Transparency and Explainability in Adaptive Systems through User Modeling Grounded in Psychological Theory*.
- Kowald, D., S. C. Pujari, and E. Lex. 2017b. "Temporal effects on hashtag reuse in twitter: A cognitive-inspired hashtag recommendation approach". In: *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee. 1401–1410.
- Kowald, D., M. Schedl, and E. Lex. 2020b. "The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study". In: *European Conference on Information Retrieval*. Springer. 35–42.
- Kowald, D., P. Seitlinger, S. Kopeinik, T. Ley, and C. Trattner. 2013. "Forgetting the words but remembering the meaning: Modeling forgetting in a verbal and semantic tag recommender". In: *Mining, Modeling, and Recommending 'Things' in Social Media*. Springer. 75–95.
- Kowald, D., P. Seitlinger, C. Trattner, and T. Ley. 2014. "Long time no see: The probability of reusing tags as a function of frequency and recency". In: *Proceedings of the 23rd International Conference on World Wide Web*. 463–468.
- Kruschke, J. K. 1992. "ALCOVE: an exemplar-based connectionist model of category learning." *Psychological review*. 99(1): 22.
- Kuan, H.-H., G.-W. Bock, and J. Lee. 2007. "A cognitive dissonance perspective of customers' online trust in multi-channel retailers". In: *European Conference on Information Systems*.
- Lamming, M. and M. Flynn. 1994. "Forget-me-not: Intimate computing in support of human memory". In: *Proc. FRIEND21, 1994 Int. Symp. on Next Generation Human Interface*. Vol. 4. Citeseer.
- Leake, D. 2001. "Problem Solving and Reasoning: Case-based". In: *International Encyclopedia of the Social & Behavioral Sciences*. Ed. by N. J. Smelser and P. B. Baltes. Oxford: Pergamon. 12117–12120. ISBN: 978-0-08-043076-8. doi: <https://doi.org/10.1016/B0-08-043076-7/00545-3>. URL: <http://www.sciencedirect.com/science/article/pii/B0080430767005453>.
- Leake, D. 2015. "Problem solving and reasoning: Case-based". *Int. Encycl. Soc. Behav. Sci*: 56–60.
- Lex, E., D. Kowald, and M. Schedl. 2020. "Modeling Popularity and Temporal Drift of Music Genre Preferences". *Transactions of the International Society for Music Information Retrieval*. 3(1).
- Ling, K., G. Beenen, P. Ludford, X. Wang, K. Chang, X. Li, D. Cosley, D. Frankowski, L. Terveen, A. M. Rashid, et al. 2005. "Using social psychology to motivate contributions to online communities". *Journal of Computer-Mediated Communication*. 10(4): 00–00.
- Lipton, Z. C. 2018. "The mythos of model interpretability". *Queue*. 16(3): 31–57.
- Liu, X., A. Datta, and E.-P. Lim. 2014. *Computational trust models and machine learning*. CRC Press.
- Loewenstein, G. and J. S. Lerner. 2003. "The role of affect in decision making". *Handbook of affective science*. 619(642): 3.
- Lops, P., D. Jannach, C. Musto, T. Bogers, and M. Koolen. 2019. "Trends in content-based recommendation". *User Modeling and User-Adapted Interaction*. 29(2): 239–249.
- Lorenzi, F. and F. Ricci. 2003. "Case-based recommender systems: A unifying view". In: *IJCAI Workshop on Intelligent Techniques for Web Personalization*. Springer. 89–113.
- Love, B. C., D. L. Medin, T. M. Gureckis, B. C. Love, T. M. Gureckis, and D. O. Psychology. 2004. "SUSTAIN: A Network Model of Category Learning". *Psychological Review*: 309–332.
- Lu, F. and N. Tintarev. 2018. "A Diversity Adjusting Strategy with Personality for Music Recommendation". In: *Proceedings of the 5th Joint Workshop on Interfaces and Human Decision Making for Recommender Systems, co-located with ACM Conference on Recommender Systems (RecSys 2018)*. 7–14.
- Maanen, L. V. and J. N. Marewski. 2009. "Recommender systems for literature selection: A competition between decision making and memory models". In: *Proceedings of the 31st Annual Meeting of the Cognitive Science Society: Cognitive Science Society*. 2914–2919.
- Mandl, M., A. Felfernig, E. Teppan, and M. Schubert. 2011. "Consumer decision making in knowledge-based recommendation". *Journal of Intelligent Information Systems*. 37(1): 1–22.
- Masthoff, J. 2004a. "Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers". *User Modeling and User-Adapted Interaction*. 14(1): 37–85.
- Masthoff, J. 2004b. "Group modeling: Selecting a sequence of television items to suit a group of viewers". In: *Personalized digital television*. Springer. 93–141.
- Masthoff, J. 2005. "The pursuit of satisfaction: affective state in group recommender systems". In: *International Conference on User Modeling*. Springer. 297–306.
- Masthoff, J. 2011. "Group recommender systems: Combining individual models". In: *Recommender systems handbook*. Springer. 677–702.
- Masthoff, J. 2015. "Group recommender systems: aggregation, satisfaction and group attributes". In: *Recommender Systems Handbook*. Springer. 743–776.
- Masthoff, J. and A. Gatt. 2006. "In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems". *User Modeling and User-Adapted Interaction*. 16(3-4): 281–319.
- Matlin, M. and D. Stang. 1978. "Pollyanna principle". *Psychology Today*. 11(10): 56.
- McCrae, R. R. and O. P. John. 1992. "An Introduction to the Five-Factor Model and Its Applications". *Journal of Personality*. 60(2): 175–215. doi: <https://doi.org/10.1111/j.1467-6494.1992.tb00970.x>.
- McCroskey, J. C. et al. 1984. "Self-report measurement". *Avoiding communication: Shyness, reticence, and communication apprehension*: 81–94.

- McGinty, L. and J. Reilly. 2011. "On the evolution of critiquing recommenders". In: *Recommender Systems Handbook*. Springer. 419–453.
- McGuire, W. J. 1969. "The nature of attitudes and attitude change". In: ed. by E. A. bibinitperiod G. Lindzey. 2nd. Massachusetts: Addison-Wesley. 136–314.
- McNee, S. M., S. K. Lam, J. A. Konstan, and J. Riedl. 2003. "Interfaces for eliciting new user preferences in recommender systems". In: *International Conference on User Modeling*. Springer. 178–187.
- McNee, S. M., J. Riedl, and J. A. Konstan. 2006a. "Being accurate is not enough: how accuracy metrics have hurt recommender systems". In: *CHI'06 extended abstracts on Human factors in computing systems*. 1097–1101.
- McNee, S. M., J. Riedl, and J. A. Konstan. 2006b. "Making recommendations better: an analytic model for human-recommender interaction". In: *CHI'06 extended abstracts on Human factors in computing systems*. 1103–1108.
- McSherry, D. 2005. "Explanation in recommender systems". *Artificial Intelligence Review*. 24(2): 179–197.
- Mehrabian, A. 1980. *Basic dimensions for a general psychological theory: implications for personality, social, environmental, and developmental studies*. Social Environmental and Developmental Studies. Oelgeschlager, Gunn & Hain. ISBN: 9780899460048.
- Meske, C. and T. Potthoff. 2017. "The DINU-model-a process model for the design of nudges". In: *Proceedings of the 25th European Conference on Information Systems (ECIS)*.
- Milano, S., M. Taddeo, and L. Floridi. 2020. "Recommender systems and their ethical challenges". *AI & SOCIETY*. 35(4): 957–967.
- Miller, T. 2019. "Explanation in artificial intelligence: Insights from the social sciences". *Artificial Intelligence*. 267: 1–38.
- Missier, F. D. 2014. "Memory and Decision Making: From Basic Cognitive Research to Design Issues". In: *International Workshop on Decision Making and Recommender Systems (DMRS)*. 8–13.
- Mizgajski, J. and M. Morzy. 2019. "Affective recommender systems in online news industry: how emotions influence reading choices". *User Modeling and User-Adapted Interaction*. 29(2): 345–379. ISSN: 1573-1391. doi: 10.1007/s11257-018-9213-x. URL: <https://doi.org/10.1007/s11257-018-9213-x>.
- Mojzisch, A. and S. Schulz-Hardt. 2010. "Knowing other's preferences degrades the quality of group decisions". *Journal of Personality and Social Psychology*. 98(5): 794–808.
- Moraveji, N., D. Russell, J. Bien, and D. Mease. 2011. "Measuring improvement in user search performance resulting from optimal search tips". In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 355–364.
- Mozer, M. C. and R. V. Lindsey. 2016. "PREDICTING AND IMPROVING MEMORY RETENTION". *Big data in cognitive science*: 34.
- Muhammad, K., A. Lawlor, R. Rafter, and B. Smyth. 2015. "Great explanations: Opinionated explanations for recommendations". In: *International Conference on Case-Based Reasoning*. Springer. 244–258.
- Murphy, J., C. Hofacker, and R. Mizerski. 2012. "Primacy and Recency Effects on Clicking Behavior". *Computer-Mediated Communication*. 11: 522–535.
- Musto, C., G. Semeraro, P. Lops, M. De Gemmis, and G. Lekkas. 2015. "Personalized finance advisory through case-based recommender systems and diversification strategies". *Decision Support Systems*. 77: 100–111.
- Nalmpantis, O. and C. Tjortjis. 2017. "The 50/50 Recommender: A Method Incorporating Personality into Movie Recommender Systems". In: *Engineering Applications of Neural Networks*. Ed. by G. Boracchi, L. Iliadis, C. Jayne, and A. Likas. Cham: Springer International Publishing. 498–507. ISBN: 978-3-319-65172-9.
- Nanou, T., G. Lekakos, and K. Fouskas. 2010. "The effects of recommendations' presentation on persuasion and satisfaction in a movie recommender system". *Multimedia systems*. 16(4-5): 219–230.
- Neisser, U. 1967. *Cognitive Psychology*. New York: Appleton-Century-Crofts.
- Nguyen, H., J. Masthoff, and P. Edwards. 2007. "Modelling a receiver's position to persuasive arguments". *Persuasive Technology*: 271–282.
- Nguyen, T. N., F. Ricci, A. Delic, and D. G. Bridge. 2019. "Conflict resolution in group decision making: insights from a simulation study". *User Model. User Adapt. Interact.* 29(5): 895–941.
- Nguyen, T. T., F. M. Harper, L. Terveen, and J. A. Konstan. 2018. "User personality and user satisfaction with recommender systems". *Information Systems Frontiers*. 20(6): 1173–1189.
- Nunes, M. A. S. N. 2008. "Recommender systems based on personality traits". *PhD thesis*. Université Montpellier II - Sciences et Techniques du Languedoc.
- O'Brien, H. L. and E. G. Toms. 2008. "What is user engagement? A conceptual framework for defining user engagement with technology". *Journal of the American Society for Information Science and Technology*. 59(6): 938–955.
- O'Brien, H. L. and E. G. Toms. 2010. "The development and evaluation of a survey to measure user engagement". *Journal of the American Society for Information Science and Technology*. 61(1): 50–69.
- O'Connor, M., D. Cosley, J. A. Konstan, and J. Riedl. 2001. "PolyLens: A Recommender System for Groups of Users". In: *Proceedings of the Seventh Conference on European Conference on Computer Supported Cooperative Work. ECSCW'01*. Bonn, Germany: Kluwer Academic Publishers. 199–218. ISBN: 0-7923-7162-3.
- O'keefe, D. J. 2015. *Persuasion: Theory and research*. Sage Publications.
- Orellana-Rodríguez, C., E. Diaz-Aviles, and W. Nejdl. 2015. "Mining Affective Context in Short Films for Emotion-Aware Recommendation". In: *Proceedings of the 26th ACM Conference on Hypertext and Social Media. HT '15*. Guzelyurt, Northern Cyprus: ACM. 185–194. ISBN: 978-1-4503-3395-5. doi: 10.1145/2700171.2791042. URL: <http://doi.acm.org/10.1145/2700171.2791042>.

- Ormerod, T. 1990. *Psychology of Programming. Human cognition and programming. Human cognition and programming*. Academic Press. 63–82.
- Ortloff, A.-M., S. Zimmerman, D. Elsweiler, and N. Henze. 2021. “The Effect of Nudges and Boosts on Browsing Privacy in a Naturalistic Environment”. In: *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval. CHIIR '21*. Canberra ACT, Australia: Association for Computing Machinery. 63–73. ISBN: 9781450380553. doi: [10.1145/3406522.3446014](https://doi.org/10.1145/3406522.3446014). URL: <https://doi.org/10.1145/3406522.3446014>.
- Paiva, R. O. A., I. I. Bittencourt, A. P. da Silva, S. Isotani, and P. Jaques. 2015. “Improving pedagogical recommendations by classifying students according to their interactional behavior in a gamified learning environment”. In: *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. 233–238.
- Pavlik, P. I. and J. R. Anderson. 2008. “Using a model to compute the optimal schedule of practice.” *Journal of Experimental Psychology: Applied*. 14(2): 101.
- Payne, J., J. Bettman, and E. Johnson. 1993. *The Adaptive Decision Maker*. Cambridge University Press.
- Perloff, R. M. 2020. *The dynamics of persuasion: Communication and attitudes in the twenty-first century*. Routledge.
- Piazza, A., P. Kröckel, and F. Bodendorf. 2017. “Emotions and Fashion Recommendations: Evaluating the Predictive Power of Affective Information for the Prediction of Fashion Product Preferences in Cold-start Scenarios”. In: *Proceedings of the International Conference on Web Intelligence. WI '17*. Leipzig, Germany: ACM. 1234–1240. ISBN: 978-1-4503-4951-2. doi: [10.1145/3106426.3109441](https://doi.acm.org/10.1145/3106426.3109441). URL: <https://doi.acm.org/10.1145/3106426.3109441>.
- Prins, F. J., R. J. Nadolski, A. J. Berlanga, H. Drachsler, H. G. Hummel, and R. Koper. 2008. “Competence Description for Personal Recommendations: The importance of identifying the complexity of learning and performance situations”. *Journal of Educational Technology & Society*. 11(3): 141–152.
- Psychology, A. M. (O. 2012. “The Computational Metaphor and Cognitive Psychology”. *Irish Journal of Psychology*. 2(10): 143–161.
- Pu, P. and L. Chen. 2006. “Trust building with explanation interfaces”. In: *Proceedings of the 11th international conference on Intelligent user interfaces*. 93–100.
- Pu, P., L. Chen, and R. Hu. 2011. “A user-centric evaluation framework for recommender systems”. In: *Proceedings of the fifth ACM conference on Recommender systems*. 157–164.
- Pu, P., L. Chen, and R. Hu. 2012. “Evaluating recommender systems from the user's perspective: survey of the state of the art”. *User Modeling and User-Adapted Interaction*. 22(4-5): 317–355.
- Quijano-Sánchez, L., J. A. Recio-García, and B. Diaz-Agudo. 2010. “Personality and Social Trust in Group Recommendations”. In: *Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence*. Vol. 02. *ICTAI'10*. Washington, DC, USA: IEEE Computer Society. 121–126. ISBN: 978-0-7695-4263-8.
- Ranjith, N. 2012. “Serial Position Curve”. In: *Encyclopedia of the Sciences of Learning*. Ed. by N. M. Seel. Boston, MA: Springer US. 3050–3052. ISBN: 978-1-4419-1428-6. doi: [10.1007/978-1-4419-1428-6_1816](https://doi.org/10.1007/978-1-4419-1428-6_1816). URL: https://doi.org/10.1007/978-1-4419-1428-6_1816.
- Ravi, L. and S. Vairavasundaram. 2017. “Learning recency and inferring associations in location based social network for emotion induced point-of-interest recommendation”. *Journal of Information Science and Engineering*. 33(Jan.): 1629–1647. doi: [10.6688/JISE.2017.33.6.15](https://doi.org/10.6688/JISE.2017.33.6.15).
- Recio-García, J. A., G. Jimenez-Díaz, A. A. Sanchez-Ruiz, and B. Diaz-Agudo. 2009. “Personality Aware Recommendations to Groups”. In: *Proceedings of the Third ACM Conference on Recommender Systems. RecSys '09*. New York, New York, USA: ACM. 325–328. ISBN: 978-1-60558-435-5.
- Ren, L. 2015. “A Time-Enhanced Collaborative Filtering Approach”. In: *2015 4th International Conference on Next Generation Computer and Information Technology (NGCIT)*. IEEE. 7–10.
- Rentfrow, P., L. R. Goldberg, and R. Zilca. 2011. “Listening, Watching, and Reading: The Structure and Correlates of Entertainment Preferences”. *Journal of Personality*. 79(Apr.): 223–258. doi: [10.1111/j.1467-6494.2010.00662.x](https://doi.org/10.1111/j.1467-6494.2010.00662.x).
- Resnick, P., N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. 1994. “GroupLens: an open architecture for collaborative filtering of netnews”. In: *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. 175–186.
- Resnick, P. and H. R. Varian. 1997. “Recommender systems”. *Communications of the ACM*. 40(3): 56–58.
- Ricci, F., B. Arslan, N. Mirzadeh, and A. Venturini. 2002. “ITR: a case-based travel advisory system”. In: *European Conference on Case-Based Reasoning*. Springer. 613–627.
- Ricci, F., D. Cavada, N. Mirzadeh, A. Venturini, et al. 2006. “Case-based travel recommendations”. *Destination recommendation systems: behavioural foundations and applications*: 67–93.
- Ricci, F., L. Rokach, and B. Shapira. 2011. “Introduction to recommender systems handbook”. In: *Recommender systems handbook*. Springer. 1–35.
- Ricci, F., L. Rokach, B. Shapira, and P. B. Kantor, eds. 2015. *Recommender Systems Handbook*. 2nd. Springer.
- Ricci, F. and H. Werthner. 2001. “Case base querying for travel planning recommendation”. *Information Technology & Tourism*. 4(3-4): 215–226.
- Rich, E. 1989. “Stereotypes and user modeling”. In: *User models in dialog systems*. Springer. 35–51.
- Riesbeck, C. K. and R. C. Schank. 2013. *Inside case-based reasoning*. Psychology Press.
- Roese, N. J. 1997. “Counterfactual thinking”. *Psychological bulletin*. 121(1): 133.
- Rossi, S. and F. Cervone. 2016. “Social Utilities and Personality Traits for Group Recommendation: A Pilot User Study”. In: *Proceedings of the 8th International Conference on Agents and Artificial Intelligence (ICAART 2016), Volume 1, Rome, Italy, February 24-26, 2016*. Ed. by H. J. van den Herik and J. Filipe. Rome, Italy: SciTePress. 38–46.

- Russell, J. A. 1980. "A Circumplex Model of Affect". *Journal of Personality and Social Psychology*. 39(6): 1161–1178.
- Rutledge-Taylor, M. F. and R. L. West. 2007. "MALTA: Enhancing ACT-R with a Holographic Persistent Knowledge Store". In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. 1433–1438.
- Rutledge-Taylor, M. F., A. Vellino, and R. L. West. 2008. "A holographic associative memory recommender system". In: *2008 Third International Conference on Digital Information Management*. 87–92. doi: [10.1109/ICDIM.2008.4746700](https://doi.org/10.1109/ICDIM.2008.4746700).
- Saari, T., N. Ravaja, J. Laarni, and M. Turpeinen. 2005. "Towards emotionally adapted games based on user controlled emotion knobs". In: *Proceedings of DiGRA 2005 Conference*. Burnaby, BC, Canada: Simon Fraser University.
- Saari, T., N. Ravaja, J. Laarni, K. Kallinen, and M. Turpeinen. 2004a. "Towards emotionally adapted Games". *Proceedings of Presence*. 13(15,10): 2004.
- Saari, T., N. Ravaja, J. Laarni, M. Turpeinen, and K. Kallinen. 2004b. "Psychologically targeted persuasive advertising and product information in e-commerce". In: *Proceedings of the 6th international conference on Electronic commerce*. 245–254.
- Sabater-mir, J., J. Cuadros, and P. Garcia. 2013. "Towards a framework that allows using a cognitive architecture to personalize recommendations in e-commerce". In: *11th European Workshop on Multi-Agent Systems*. 3–17.
- Sarwar, B., G. Karypis, J. Konstan, and J. Riedl. 2001. "Item-based collaborative filtering recommendation algorithms". In: *Proceedings of the 10th international conference on World Wide Web*. 285–295.
- Schafer, J. B., D. Frankowski, J. Herlocker, and S. Sen. 2007. "Collaborative filtering recommender systems". In: *The adaptive web*. Springer. 291–324.
- Schäfer, T. 2016. "The Goals and Effects of Music Listening and Their Relationship to the Strength of Music Preference". *PLOS ONE*. 11(3): 1–15. doi: [10.1371/journal.pone.0151634](https://doi.org/10.1371/journal.pone.0151634). URL: <https://doi.org/10.1371/journal.pone.0151634>.
- Schedl, M. 2019. "Deep Learning in Music Recommendation Systems". *Frontiers in Applied Mathematics and Statistics*. 5: 44. ISSN: 2297-4687. doi: [10.3389/fams.2019.00044](https://doi.org/10.3389/fams.2019.00044). URL: <https://www.frontiersin.org/article/10.3389/fams.2019.00044>.
- Schedl, M., E. Gómez, E. Trent, M. Tkalcic, H. Eghbal-Zadeh, and A. Martorell. 2018. "On the Interrelation between Listener Characteristics and the Perception of Emotions in Classical Orchestra Music". *IEEE Transactions on Affective Computing*. 9(4): 507–525. ISSN: 1949-3045. doi: [10.1109/TAFFC.2017.2663421](https://doi.org/10.1109/TAFFC.2017.2663421).
- Schmuckler, M. A. 2001. "What is ecological validity? A dimensional analysis". *Infancy*. 2(4): 419–436.
- Schnabel, T., P. N. Bennett, S. T. Dumais, and T. Joachims. 2016. "Using Shortlists to Support Decision Making and Improve Recommender System Performance". In: *Proceedings of the 25th International Conference on World Wide Web*. WWW '16. Montréal, Québec, Canada: International World Wide Web Conferences Steering Committee. 987–997. ISBN: 978-1-4503-4143-1. doi: [10.1145/2872427.2883012](https://doi.org/10.1145/2872427.2883012). URL: <https://doi.org/10.1145/2872427.2883012>.
- Schneider, C., M. Weinmann, and J. Vom Brocke. 2018. "Digital nudging: guiding online user choices through interface design". *Communications of the ACM*. 61(7): 67–73.
- Schwind, C. and J. Buder. 2014. "The case for preference-inconsistent recommendations". In: *Recommender Systems for Technology Enhanced Learning*. Springer. 145–157.
- Schwind, C., J. Buder, and F. W. Hesse. 2011. "I will do it, but I don't like it: user reactions to preference-inconsistent recommendations". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 349–352.
- Seitlinger, P., D. Kowald, C. Trattner, and T. Ley. 2013. "Recommending tags with a model of human categorization". In: *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM. 2381–2386.
- Seitlinger, P. and T. Ley. 2016. "Reconceptualizing Imitation in Social Tagging: A Reflective Search Model of Human Web Interaction". In: *Proceedings of the 8th ACM Conference on Web Science*. WebSci '16. Hannover, Germany: ACM. 146–155. ISBN: 978-1-4503-4208-7. doi: [10.1145/2908131.2908157](https://doi.org/10.1145/2908131.2908157). URL: <https://doi.org/10.1145/2908131.2908157>.
- Sen, A. 1986. "Social choice theory". *Handbook of mathematical economics*. 3: 1073–1181.
- Sertkan, M., J. Neidhardt, and H. Werthner. 2019. "What is the "Personality" of a tourism destination?". *Information Technology & Tourism*. 21(1): 105–133.
- Shani, G. and A. Gunawardana. 2011. "Evaluating recommendation systems". In: *Recommender systems handbook*. Springer. 257–297.
- Sharma, R. and S. Ray. 2016. "Explanations in recommender systems: an overview". *International Journal of Business Information Systems*. 23(2): 248–262.
- Shin, D. 2020. "How do users interact with algorithm recommender systems? The interaction of users, algorithms, and performance". *Computers in Human Behavior*. 109: 106344.
- Shiv, B. and A. Fedorikhin. 1999. "Heart and mind in conflict: The interplay of affect and cognition in consumer decision making". *Journal of consumer Research*. 26(3): 278–292.
- Simon, H. A. 1966. "Theories of decision-making in economics and behavioural science". In: *Surveys of economic theory*. Springer. 1–28.
- Sinha, R. R., K. Swearingen, et al. 2001. "Comparing recommendations made by online systems and friends." *DELOS*. 106.
- Smids, J. 2012. "The voluntariness of persuasive technology". In: *International Conference on Persuasive Technology*. Springer. 123–132.

- Stanley, C. and M. D. Byrne. 2016. "Comparing Vector-Based and Bayesian Memory Models Using Large-Scale Datasets: User-Generated Hashtag and Tag Prediction on Twitter and Stack Overflow". *Psychological Methods*. 21(4): 542–565.
- Stanovich, K. E. and R. F. West. 1998. "Individual differences in rational thought." *Journal of experimental psychology: general*. 127(2): 161.
- Starke, A. D., M. C. Willemsen, and C. Snijders. 2020. "With a little help from my peers: Depicting social norms in recommender interface to promote energy conservation". In: *Proceedings of the 25th International Conference on Intelligent User Interfaces*. 568–578.
- Stettinger, M., A. Felfernig, G. Leitner, and S. Reiterer. 2015a. "Counteracting Anchoring Effects in Group Decision Making". In: *23rd Conference on User Modeling, Adaptation, and Personalization (UMAP'15) (UMAP Best Papers 2015)*. Vol. 9146. LNCS. Springer. 118–130.
- Stettinger, M., A. Felfernig, G. Leitner, S. Reiterer, and M. Jeran. 2015b. "Counteracting Serial Position Effects in the CHOICLA Group Decision Support Environment". In: *International Conference on Intelligent User Interfaces, Proceedings IUI*. Vol. 2015. 148–157. doi: [10.1145/2678025.2701391](https://doi.org/10.1145/2678025.2701391).
- Stewart, B. 2011. "Personality and play styles: A unified model". *Gamasutra, September*. 1.
- Sunstein, C. R. 2015. "The ethics of nudging". *Yale J. on Reg.* 32: 413.
- Surendren, D. and V. Bhuvaneswari. 2014. "A framework for analysis of purchase dissonance in recommender systems using association rule mining". In: *2014 International Conference on Intelligent Computing Applications*. IEEE. 153–157.
- Swearingen, K. and R. Sinha. 2002. "Interaction design for recommender systems". In: *Designing Interactive Systems*. Vol. 6. No. 12. Citeseer. 312–334.
- Teppan, E. and A. Felfernig. 2012. "Minimization of Decoy Effects in Recommender Result Sets". *Web Intelligence and Agent Systems*. 1(4): 385–395.
- Teppan, E. C. and A. Felfernig. 2009a. "Calculating decoy items in utility-based recommendation". In: *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer. 183–192.
- Teppan, E. C. and A. Felfernig. 2009b. "Impacts of decoy elements on result set evaluations in knowledge-based recommendation". *International Journal of Advanced Intelligence Paradigms*. 1(3): 358–373.
- Teppan, E. C. and M. Zanker. 2015. "Decision biases in recommender systems". *Journal of Internet Commerce*. 14(2): 255–275.
- Thaker, K., Y. Huang, P. Brusilovsky, and H. Daqing. 2018. "Dynamic knowledge modeling with heterogeneous activities for adaptive textbooks". In: *The 11th International Conference on Educational Data Mining*. 592–595.
- Thaler, R. 1980. "Toward a positive theory of consumer choice". *Journal of economic behavior & organization*. 1(1): 39–60.
- Thaler, R. H. and C. R. Sunstein. 2009. *Nudge: Improving decisions about health, wealth, and happiness*. Penguin.
- Thaler, R. H., C. R. Sunstein, and J. P. Balz. 2013. "Choice architecture". *The behavioral foundations of public policy*: 428–439.
- Thomas, K. W. and R. H. Kilmann. 1974. *Thomas-Kilmann Conflict Mode Instrument*. PO Box 10096, California, USA: Consulting Psychologists Press.
- Thomas, K. W. 1992. "Conflict and conflict management: Reflections and update". *Journal of organizational behavior*: 265–274.
- Tintarev, N., M. Dennis, and J. Masthoff. 2013. "Adapting Recommendation Diversity to Openness to Experience: A Study of Human Behaviour". In: *User Modeling, Adaptation, and Personalization*. Ed. by S. Carberry, S. Weibelzahl, A. Micarelli, and G. Semeraro. Berlin, Heidelberg: Springer Berlin Heidelberg. 190–202. ISBN: 978-3-642-38844-6.
- Tintarev, N. and J. Masthoff. 2012. "Evaluating the effectiveness of explanations for recommender systems". *User Modeling and User-Adapted Interaction*. 22(4-5): 399–439.
- Tintarev, N. and J. Masthoff. 2015. "Explaining recommendations: Design and evaluation". In: *Recommender systems handbook*. Springer. 353–382.
- Tkalcic, M. and L. Chen. 2015a. "Personality and Recommender Systems". In: *Recommender Systems Handbook*. Ed. by F. Ricci, L. Rokach, and B. Shapira. Springer. 715–739. doi: [10.1007/978-1-4899-7637-6_21](https://doi.org/10.1007/978-1-4899-7637-6_21). url: https://doi.org/10.1007/978-1-4899-7637-6%5C_21.
- Tkalcic, M. and L. Chen. 2015b. "Personality and recommender systems". In: *Recommender systems handbook*. Springer. 715–739.
- Tkalcic, M., A. Kosir, and J. Tasic. 2011. "Affective recommender systems: the role of emotions in recommender systems". In: *Proceedings of the RecSys 2011 Workshop on Human Decision Making in Recommender Systems and User-Centric Evaluation of Recommender Systems and Their Interfaces*. Citeseer. 9–13.
- Tran, T. N. T., M. Atas, V. M. Le, R. Samer, and M. Stettinger. 2020. "Social Choice-based Explanations: An Approach to Enhancing Fairness and Consensus Aspects," *J. UCS*. 26(3): 402–431.
- Tran, T. N. T., M. Atas, A. Felfernig, V. M. Le, R. Samer, and M. Stettinger. 2019a. "Towards social choice-based explanations in group recommender systems". In: *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*. 13–21.
- Tran, T. N. T., M. Atas, A. Felfernig, R. Samer, and M. Stettinger. 2018. "Investigating Serial Position Effects in Sequential Group Decision Making". In: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. 239–243.
- Tran, T. N. T., A. Felfernig, V. M. Le, M. Atas, M. Stettinger, and R. Samer. 2019b. "User Interfaces for Counteracting Decision Manipulation in Group Recommender Systems". In: *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*. 93–98.

- Trattner, C., D. Kowald, P. Seitlinger, S. Kopeinik, and T. Ley. 2016. "Modeling activation processes in human memory to predict the reuse of tags". *The Journal of Web Science*. 2.
- Turland, J., L. Coventry, D. Jeske, P. Briggs, and A. van Moorsel. 2015. "Nudging towards security: Developing an application for wireless network selection for android phones". In: *Proceedings of the 2015 British HCI conference*. 193–201.
- Turpeinen, M. and T. Saari. 2004. "System architecture for psychological customization of communication technology". In: *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the IEEE*. 10–pp.
- Tversky, A. 1977. "Features of similarity." *Psychological review*. 84(4): 327.
- Tversky, A. and D. Kahneman. 1974. "Judgment under uncertainty: Heuristics and biases". *science*. 185(4157): 1124–1131.
- Tversky, A. and D. Kahneman. 1981. "The framing of decisions and the psychology of choice". *science*. 211(4481): 453–458.
- Tversky, A. and D. Kahneman. 1992. "Advances in prospect theory: Cumulative representation of uncertainty". *Journal of Risk and uncertainty*. 5(4): 297–323.
- Ullman, J. B. and P. M. Bentler. 2003. "Structural equation modeling". *Handbook of psychology*: 607–634.
- Wang, C.-S. and H.-L. Yang. 2012. "A recommender mechanism based on case-based reasoning". *Expert Systems with Applications*. 39(4): 4335–4343.
- Wang, S., K. Kirillova, and X. Lehto. 2017. "Reconciling unsatisfying tourism experiences: Message type effectiveness and the role of counterfactual thinking". *Tourism Management*. 60: 233–243.
- Waugh, N. C. and D. A. Norman. 1965. "Primary memory." *Psychological review*. 72(2): 89.
- Willemsen, M. C., M. P. Graus, and B. P. Knijnenburg. 2016. "Understanding the role of latent feature diversification on choice difficulty and satisfaction". *User Modeling and User-Adapted Interaction*. 26(4): 347–389.
- Wu, W., L. Chen, and L. He. 2013. "Using Personality to Adjust Diversity in Recommender Systems". In: *Proceedings of the 24th ACM Conference on Hypertext and Social Media. HT '13*. Paris, France: ACM. 225–229. ISBN: 978-1-4503-1967-6. doi: [10.1145/2481492.2481521](https://doi.acm.org/10.1145/2481492.2481521). URL: <http://doi.acm.org/10.1145/2481492.2481521>.
- Wu, W., L. Chen, Q. Yang, and Y. Li. 2019. "Inferring Students' Personality from Their Communication Behavior in Web-based Learning Systems". *International Journal of Artificial Intelligence in Education*. 29(2): 189–216.
- Wu, W., L. Chen, and Y. Zhao. 2018. "Personalizing recommendation diversity based on user personality". *User Modeling and User-Adapted Interaction*. 28(3): 237–276. issn: 1573-1391. doi: [10.1007/s11257-018-9205-x](https://doi.org/10.1007/s11257-018-9205-x). URL: <https://doi.org/10.1007/s11257-018-9205-x>.
- Xiao, B. and I. Benbasat. 2007. "E-commerce product recommendation agents: Use, characteristics, and impact". *MIS quarterly*: 137–209.
- Xu, J., X. He, and H. Li. 2020. "Deep Learning for Matching in Search and Recommendation". *Foundations and Trends® in Information Retrieval*. 14(2–3): 102–288. issn: 1554-0669. doi: [10.1561/1500000076](https://doi.org/10.1561/1500000076). URL: <http://dx.doi.org/10.1561/1500000076>.
- Yago, H., J. Clemente, and D. Rodriguez. 2018. "Competence-based recommender systems: a systematic literature review". *Behaviour & Information Technology*. 37(10-11): 958–977.
- Yang, H.-L. and C.-S. Wang. 2009. "Recommender system for software project planning one application of revised CBR algorithm". *Expert Systems with Applications*. 36(5): 8938–8945.
- Yang, H.-C. and Z.-R. Huang. 2019. "Mining personality traits from social messages for game recommender systems". *Knowledge-Based Systems*. 165: 157–168. issn: 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2018.11.025>. URL: <http://www.sciencedirect.com/science/article/pii/S095070511830577X>.
- Yang, Z., J. He, and S. He. 2019. "A collaborative filtering method based on forgetting theory and neural item embedding". In: *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. IEEE. 1606–1610.
- Yannakakis, G. N. and J. Hallam. 2011. "Ranking vs. preference: a comparative study of self-reporting". In: *International conference on affective computing and intelligent interaction*. Springer. 437–446.
- Yoo, K.-H. and U. Gretzel. 2011. "Creating more credible and persuasive recommender systems: The influence of source characteristics on recommender system evaluations". In: *Recommender systems handbook*. Springer. 455–477.
- Yoo, K.-H., U. Gretzel, and M. Zanker. 2012. *Persuasive recommender systems: conceptual background and implications*. Springer Science & Business Media.
- Yu, H. and Z. Li. 2010. "A collaborative filtering method based on the forgetting curve". In: *2010 International Conference on Web Information Systems and Mining*. Vol. 1. IEEE. 183–187.
- Zajonc, R. B. 1968. "Attitudinal effects of mere exposure." *Journal of personality and social psychology*. 9(2p2): 1.
- Zhang, J. 2011. "Anchoring effects of recommender systems". In: *Proceedings of the fifth ACM conference on Recommender systems*. 375–378.
- Zhang, S., L. Yao, A. Sun, and Y. Tay. 2019a. "Deep Learning Based Recommender System". *ACM Computing Surveys*. 52(1): 1–38. issn: 0360-0300. doi: [10.1145/3285029](https://doi.org/10.1145/3285029). URL: <http://dx.doi.org/10.1145/3285029>.
- Zhang, S., L. Yao, A. Sun, and Y. Tay. 2019b. "Deep learning based recommender system: A survey and new perspectives". *ACM Computing Surveys (CSUR)*. 52(1): 1–38.
- Zhang, Y., X. Chen, et al. 2020. "Explainable Recommendation: A Survey and New Perspectives". *Foundations and Trends® in Information Retrieval*. 14(1): 1–101.

- Zhao, L., J. Huang, and N. Zhong. 2014. "A Context-Aware Recommender System with a Cognition Inspired Model". In: *Rough Sets and Knowledge Technology*. Cham: Springer International Publishing. 613–622. ISBN: 978-3-319-11740-9.
- Zheng, L., C.-T. Lu, L. He, S. Xie, H. He, C. Li, V. Noroozi, B. Dong, and S. Y. Philip. 2019. "MARS: Memory attention-aware recommender system". In: *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE. 11–20.
- Zheng, Y. 2013. "The Role of Emotions in Context-aware Recommendation". In: *Decisions@RecSys*. 21–28.
- Ziegler, C.-N., S. M. McNee, J. A. Konstan, and G. Lausen. 2005. "Improving recommendation lists through topic diversification". In: *Proceedings of the 14th international conference on World Wide Web*. 22–32.
- Zimmerman, S., A. Thorpe, J. Chamberlain, and U. Kruschwitz. 2020. "Towards search strategies for better privacy and information". In: *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval*. 124–134.

P5 Integrating the ACT-R Framework and Collaborative Filtering for Explainable Sequential Music Recommendation (2023)
Transparency and Cognitive Models in Recommender Systems

[P5] Moscati, M., Wallmann, C., Reiter-Haas, M., **Kowald, D.**, Lex, E., Schedl, M. (2023). Integrating the ACT-R Framework and Collaborative Filtering for Explainable Sequential Music Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems (RecSys'2023)*, pp. 840–847.
DOI: <https://doi.org/10.1145/3604915.3608838>



Integrating the ACT-R Framework with Collaborative Filtering for Explainable Sequential Music Recommendation

Marta Moscati

marta.moscati@jku.at

Institute of Computational Perception,
Johannes Kepler University Linz
Linz, Austria

Christian Wallmann

ch.wallmann@welser.com

Welser Profile GmbH
Gresten, Austria

Markus Reiter-Haas

reiter-haas@tugraz.at

Graz University of Technology
Graz, Austria

Dominik Kowald

dkowald@know-center.at
Know-Center GmbH and Graz
University of Technology
Graz, Austria

Elisabeth Lex

elisabeth.lex@tugraz.at
Graz University of Technology
Graz, Austria

Markus Schedl

markus.schedl@jku.at

Institute of Computational Perception,
Johannes Kepler University Linz and
Human-centered AI Group, AI Lab,
Linz Institute of Technology
Linz, Austria

ABSTRACT

Music listening sessions often consist of sequences including repeating tracks. Modeling such relistening behavior with models of human memory has been proven effective in predicting the next track of a session. However, these models intrinsically lack the capability of recommending novel tracks that the target user has not listened to in the past. Collaborative filtering strategies, on the contrary, provide novel recommendations by leveraging past collective behaviors but are often limited in their ability to provide explanations. To narrow this gap, we propose four hybrid algorithms that integrate collaborative filtering with the cognitive architecture ACT-R. We compare their performance in terms of accuracy, novelty, diversity, and popularity bias, to baselines of different types, including pure ACT-R, kNN-based, and neural-networks-based approaches. We show that the proposed algorithms are able to achieve the best performances in terms of novelty and diversity, and simultaneously achieve a higher accuracy of recommendation with respect to pure ACT-R models. Furthermore, we illustrate how the proposed models can provide explainable recommendations.

KEYWORDS

Adaptive Control Thought-Rational (ACT-R), Collaborative Filtering, Sequential Recommendation, Music Recommender Systems, Psychology-Informed Recommender Systems, Explainability

ACM Reference Format:

Marta Moscati, Christian Wallmann, Markus Reiter-Haas, Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2023. Integrating the ACT-R Framework with Collaborative Filtering for Explainable Sequential Music Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23)*, September 18–22, 2023, Singapore, Singapore. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3604915.3608838>



This work is licensed under a [Creative Commons Attribution-NonCommercial International 4.0 License](#).

RecSys '23, September 18–22, 2023, Singapore, Singapore

© 2023 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0241-9/23/09.

<https://doi.org/10.1145/3604915.3608838>

'23), September 18–22, 2023, Singapore, Singapore. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3604915.3608838>

1 INTRODUCTION

Music is often consumed sequentially. Therefore, music recommendation [46, 48] is often formulated as a session completion task: tracks should be recommended to a user according to their interactions in the recent past, i.e., those within the current session. The most effective recommender systems (RSs) for sequential recommendation are based on collaborative filtering (CF) [11, 14, 32, 33]. These algorithms provide recommendations according to past collective user behavior. Although effective, the recommendations provided by CF algorithms are often hard to justify, either due to the model architecture or the complexity of the data they base their recommendations on. Another major distinguishing characteristic of music RSs compared to general RSs is that music listeners often listen to tracks they already listened to in the past [7, 42, 48]. This observation served as a basis for translating cognitive architectures, i.e., models of the structure of human mind, to the domain of RSs, and evaluate their effectiveness in predicting users' relistening behaviors. In particular, the memory module of the *Adaptive Control of Thought—Rational* (ACT-R) cognitive architecture [6, 44] has been proven effective in predicting which tracks the user will relisten to, based on the tracks listened to in the past [40]. However, despite their effectiveness in modeling user's relistening behavior, leveraging these models based on ACT-R for sequential music recommendation does not allow recommending *novel* tracks, i.e., tracks the target user has never interacted with before. To compensate for these shortcomings, we design four algorithms that integrate ACT-R with CF. Since each component of the memory module of ACT-R is designed to model a different aspect of human memory, the recommendations provided by the proposed algorithms are explainable. We measure the performance of the proposed RSs in terms of accuracy, novelty, diversity, and popularity bias of the recommended tracks since these are all aspects that affect the user's satisfaction with the system [16, 48]. Additionally, we show how the explainability of the proposed algorithms can be advantageous in a

multistakeholder RS [1], concerning end users, platform providers, and content producers.

In summary, this work provides the following contributions to the RS domain: (1) We propose four algorithms that integrate various components of the cognitive architecture ACT-R with CF for sequential recommendation. (2) We provide an extensive analysis of the performance of these algorithms by performing experiments on the LFM-2b dataset [45] of Last.fm listening logs. We compare the performance of the algorithms with well-established baselines, including algorithms that solely rely on the cognitive architecture ACT-R, on k -nearest-neighbors (kNN), and on deep neural networks (DNNs). Our experiments show that the proposed algorithms increase the novelty and diversity of recommendations compared to the baselines. Moreover, we find that the hybrid approaches outperform pure ACT-R models in terms of accuracy. (3) We exemplify how the proposed algorithms can be used to explain music recommendations.

2 BACKGROUND AND RELATED WORK

In the following, we briefly present work on sequential recommendation and on RSs based on cognitive architectures, thereby introducing the fundamentals for the proposed algorithms.

2.1 Sequential Recommendation

Some of the most successful sequential RSs leverage the similarity of the initial segment of the session to be completed to other sessions. Extensions of these algorithms also introduce *temporal reweighting*, i.e. they consider factors that model the position and recency of the interactions with the items [11, 31, 34, 41]. For instance, Ludewig et al. [34] reweight the recommendation score as follows: if an item i appeared at position t_i , its relevance as a recommendation for position t_{ref} is weighted by a factor given by $w_i = (t_{\text{ref}} - t_i)^{-d}$, where t_{ref} stands for the timestamp of the next track, i.e., the one the algorithm is aiming to predict. Some effective approaches use DNN architectures for sequential recommendation [8, 13, 17, 27, 30, 49, 50]. Finally, other works model the sessions by representing them as graphs, and leverage graph neural networks [5, 12, 38, 50–52].

2.2 Music Recommender Systems

Sequential RSs are particularly relevant in the context of music recommendation [47, 48] since they address tasks such as next-track recommendation or automatic playlist continuation. For an overview of the approaches used for sequential music recommendation we refer the reader to Quadrana et al. [39]. Additionally, since providing explanations for the recommendations can positively impact the users' trust and engagement, the interest in addressing explainability in the context of music RSs has been increasing in the last years; for an overview of the topic we refer the reader to Afchar et al. [4].

2.3 Cognition-inspired Recommender Systems

Cognition-inspired RSs use models from the domain of cognitive psychology to create RSs, often using theories of human memory [26, 28]. Our work focuses on ACT-R [6, 44]. Several studies leveraged the memory module of ACT-R for tasks such as hashtag recommendation [19], item recommendation in social tagging systems [21], next genre prediction [25], next artist prediction [18], job recommendation [20, 22], or predicting mobile app usages [53].

In particular, Reiter-Haas et al. [40] use ACT-R's memory module for completing music streaming sessions. The components of the module are described in the remainder of this section.

Base-Level Learning (BLL): The BLL component captures the tendency of human memory of favoring instances that occurred frequently and recently in the past. Similar to Reiter-Haas et al. [40], given the timestamp t_{ref} of the next track in the session, i.e., the one the algorithm is aiming to predict, and given an item i , we define its BLL activation as $B_i = \sum_{j=1}^n (t_{\text{ref}} - t_{ij})^{-d}$. The sum extends to all the n past interactions of the user with item i , and t_{ij} stands for the timestamp of the j^{th} interaction with item i .

Spreading (S): The spreading component favors items that occur frequently in the current *context*. In agreement with how context is defined within the ACT-R cognitive architecture, Reiter-Haas et al. [40] define the context as the last item the user interacted with. This component hence tends to favor items that the user often interacted with in sessions that contain the most recent item in the sequence. The corresponding activation is given by $S_i = \frac{P(i \in C_k)}{P(i)}$ [10, 40], where item k is the last item of the sequence, and $P(i)$ and $P(i \in C_k)$ stand for the probabilities that track i appears in any session, and in a session containing item k , respectively.

Partial Matching (PM): The PM component [40] aims at favoring items that are *similar* to the context item k , i.e., the last item the user interacted with. The corresponding activation is given by $P_i = \text{sim}(i, k)$, where $\text{sim}(i, k)$ represents the similarity between item i and the context item k . Assuming an item i to be represented by a feature vector f_i , the similarity $\text{sim}(i, k)$ between i and k is defined as the scalar product of the corresponding feature vectors, $\text{sim}(i, k) = f_i \cdot f_k$.

Valuation (V): The valuation component [15, 40] aims at measuring the *value* attributed by a user to an item. The corresponding activation for an item i with which the user interacted n times is defined iteratively as $V_i(n) = V_i(n-1) + \alpha (R_i(n) - V_i(n-1))$, where $R_i(n)$ is the reward assigned to item i for the n^{th} interaction. The starting valuation is set to $V_i(0) = 0$ for all tracks, and the learning rate α is considered as a hyperparameter. In the context of sequential music recommendation [40], the reward $R_i(j)$ is typically either binary, i.e., $R_i(j) = 1 \forall j \in [1, \dots, n]$, or given by the duration of the j^{th} interaction with respect to the total track length.

Noise (N): The noise component models aspects of randomness in the user's behavior. The corresponding activation is given by $\epsilon_i = \text{rng}()$, where $\text{rng}()$ is a random number generator.

Reiter-Haas et al. [40] show that ACT-R-based approaches outperform baselines such as algorithms selecting the most recent track, in terms of accuracy of predictions. Compared to their work, we integrate ACT-R and CF, extend the analysis to beyond-accuracy metrics, and provide a comparison with more recent baselines. Finally, we also leverage ACT-R for explaining the recommendations.

3 METHODS

To integrate ACT-R and CF for sequential music recommendation, we propose the following hybrid algorithms.

Social ACT-R Kowald et al. [19] propose an algorithm for hashtag recommendation that combines the ACT-R activations of the target user's past hashtags with the ACT-R activations of the target user's followees.

We adapt this strategy to the music domain. In order to include the listening behavior of other users, we first define the target user's "followees" as the set of k users that are most similar to the target user. The similarity $\text{sim}_{\text{ACT-R}}(u, j)$ between the target user u and another user j is computed as cosine similarity between the vector representing their listening events, i.e., their interactions with tracks, reweighted with the ACT-R activations. The value of the social component (SC) assigned to track i for session u is then defined as a similarity-weighted average of the ACT-R activations of the k followees, $SC_i = \sum_{j \leq k} \text{ACT-R}(j, i) \cdot \text{sim}_{\text{ACT-R}}(u, j)$. The SC and the target user's ACT-R activation of the track are normalized by applying softmax over all tracks and added up to obtain the final recommendation score.

ACT-R + BPR This model extends ACT-R with a component that favors tracks that have a similar interaction history to the one of the context track, i.e., the last track the user listened to. For this purpose, we pretrain a matrix factorization RS with Bayesian personalized ranking (BPR) [43]. Each track is mapped to their BPR embedding v_i . We then compute the similarity $\text{sim}_{\text{BPR}}(i, j)$ between two tracks i, j as cosine similarity between their BPR embeddings v_i, v_j . The recommendation score of a track i is obtained by adding up the softmax-normalized BPR similarity $\text{sim}_{\text{BPR}}(i, k)$ with the context item k and the softmax-normalized target user's ACT-R activation of i . In addition, we consider a version of this model in which only the similarity $\text{sim}_{\text{BPR}}(i, k)$ between the BPR embeddings is considered when computing the recommendation score. This model is referred to as Item BPR.

Weighted MultVAE: We integrate ACT-R with MultVAE [29], since this model allows providing recommendations to users that are not in the train set, and since it provides accurate recommendations in several domains, including music [9, 36]. We pretrain and optimize an instance of MultVAE. We then reweight the components of the vector representing the listening events of the target user u either with the ACT-R activations or with the temporal reweighting factor w_i (see Section 2.1), converting it to a vector of ratings. We feed this vector to the pretrained MultVAE and perform a forward pass of MultVAE to select the tracks to recommend.

Weighted UserkNN: Similar to Weighted MultVAE, we first train an instance of MultVAE and then perform a forward pass on the temporally reweighed vector representing the listening events of the target user, extracting the latent representations l_u of the target user u encoded by MultVAE. We encode the binarized¹ profile of the other users in the dataset and select the k users with latent representations having the largest cosine similarity $\text{sim}_{\text{MultVAE}}(u, j)$ to the latent representation l_u of the target user. We take the weighted average of the binarized profiles of the k nearest users, using the similarity of the latent representations as weights for the weighted average, as recommendation score. The score of track i is therefore given by $\sum_{j \leq k} r(j, i) \cdot \text{sim}_{\text{MultVAE}}(u, j)$, where $r(j, i)$ represents the binarized interaction of user j with item i .

¹In agreement with the reweighing proposed by Ludewig et al. [34], we do not apply temporal reweighing to the profile of the non-target users.

4 EXPERIMENTAL SETUP

In this section, we describe the setup for our experiments, i.e., the baseline models, the evaluation metrics, the dataset, as well as the training and hyperparameter selection.

4.1 Baselines and underlying models

We compare the performance of the approaches introduced in Section 3 to those of two models effective in the task of sequential recommendation – GRU4Rec [49] and temporal UserkNN [32, 33] – and two models effective in predicting relistening behavior – MostRecent [40] and ACT-R [40].

GRU4Rec: This algorithm makes use of recurrent neural networks for sequential recommendation [49]. We take it as DNN-based baseline since it is among the DNN approaches achieving high accuracy, large dataset coverage, and low popularity bias, simultaneously [33].

Temporal UserkNN: Models including a temporal reweighting (see Section 2.1) are competitive with DNN-based approaches in terms of accuracy [33, 34]. In including this class of models as baselines, we reweight the vectors representing the listening events of the target user, as well as those representing the other users, as described in Section 2.1. We then compute the cosine similarity of the resulting vectors. The reweighted interactions of the k nearest users are averaged according to the similarity to the target user and used as recommendation scores.

MostRecent: This algorithm recommends the most recent tracks in the sequence, and has been proven effective in predicting users' relistening behavior [40], especially in accurately predicting the next track in the session (see discussion of Next-HR in Section 4.2).

ACT-R: This model corresponds to the one used by Reiter-Haas et al. in [40] for modeling the users' music relistening behavior; we refer the reader to Section 2.3 for the description of the individual components.²

4.2 Evaluation metrics

The performance of the algorithms is evaluated on the task of rolling session completion. Similar to Reiter-Haas et al. [40], for each target user we shift a sliding window of one week with a hop size of one listening event and define sessions as sequences of listening events without gaps of more than 30 minutes between consecutive tracks. Given a target user's session of N tracks and the target user's listening events of the previous seven days, we assume a session segment of length $l < N$ to be known and predict the remaining $N - l$ tracks in the session. For each session, we consider all possible initial segment lengths, $l = 1, \dots, N - 1$.

Accuracy: We include two metrics for the accuracy of recommendations. Since ACT-R can only recommend items that the user already listened to, if the number of past interactions is less than the number of tracks in the remainder of the sessions, i.e., those to provide recommendations for, the algorithm will not be able to provide recommendations for the full session. This results in a higher precision and a lower recall. To mitigate this effect, we

²Similar to Reiter-Haas et al. [40], we normalize each component by applying softmax over all tracks and add up the results to obtain the ACT-R activation of a track. For all ACT-R-based models, preliminary experiments showed that including ϵ and PM based on different versions of the features provided by Spotify reduces the performance of the algorithms. We, therefore, omit them. Based on preliminary experiments, we also set $d = 0.5$.

combine the precision and recall of recommendations into the F_1 score. Similar to Reiter-Haas et al. [40], we also evaluate the *next hit rate* (Next-HR), i.e., the ability of the algorithm to correctly predict the next track of the session.

Novelty: The novelty of recommendations is measured as the fraction of recommended tracks that have not been listened to by the target user. In addition, to evaluate the quality of novel recommendations, we also report the precision of novel recommendations, P-Noveltiy.

Diversity: The diversity of the recommendations is measured with respect to the genres.³ Since a higher diversity should indicate that the recommended tracks belong to different genres, we define diversity as the Shannon entropy of the distribution of genres over the recommended tracks.

Popularity bias: To evaluate the tendency of the algorithms to overrepresent popular tracks compared to the ones in the user's past listening events, we compute the Jensen-Shannon divergence between the popularity distribution of tracks in the user's past listening events, and over the recommended tracks [23]. A high popularity bias thus indicates that recommended tracks are more popular than those already listened to by the target user.⁴

4.3 Dataset, training, and evaluation

Similar to Melchiorre et al. [35], we conduct our experiments on the extract of the large LFM-2b dataset [45]⁵ corresponding to the last month (20/02/2020 - 19/03/2020) and remove users that listened to more tracks than the 99th percentile of all users. We apply 10-core filtering to users and items and split each user's listening events temporally in a 60% train, 20% validation, and a 20% test set. The resulting dataset consists of 2 889 028 listening events, 12 679 users and 101 837 items. The 60% train set is used to determine the similarity for approaches relying on kNN, and for training and selecting the best hyperparameters of GRU4Rec, BPR, and MultVAE. For GRU4Rec, the most recent 20% interactions of each user in the 60% train set are used for selecting the best hyperparameter configuration on a grid space based on that reported by Ludewig et al. [33, 34]. The optimization of the BPR and MultVAE instances required by ACT-R + BPR, Weighted MultVAE, and Weighted MultVAE-UserkNN is performed previously to the optimization of the algorithms that rely on them, and therefore on a separate set: the 60% train set is converted to a binarized version (i.e., $b_{ui} = 1 \iff u$ listened to i at least once) and 20% randomly selected binarized interactions of each user are used for selecting the hyperparameter configuration achieving the highest NDCG@10 on a grid space based on that reported by Melchiorre et al. [36]. The 20% validation set is used to select the best configuration of kNN-, temporal, and ACT-R-based algorithms described in Sections 3 and 4.1, on a grid space based on that reported by Ludewig et al. [33, 34] and Reiter-Haas et al. [40],

³The track genre is assigned based on the Last.fm tags of the track, selecting the genre with the highest tag weight. We use the list of Discogs genres, available at https://mtg.github.io/acousticbrainz-genre-dataset/data_stats/ as possible genres of a track.

⁴The popularity of a track is defined as the ratio of the total number of listening events it accounts for [3, 24]. The distributions are computed over popularity classes, each defined in terms of percentiles: after sorting tracks according to the number of listening events, popular-, mid-, and niche-tracks account for 20, 60, and 20% of all events, respectively.

⁵<http://www.cp.jku.at/datasets/LFM-2b/>

selecting the configuration achieving the highest F_1 score. Since the average number of session completion tasks per user is 67, providing recommendations for the sessions of all 12 679 users would result in roughly 850 000 session recommendations, i.e., roughly 850 000 test users in a standard recommendation scenario. Therefore, to reduce computational costs, we randomly sample 100 users and evaluate the algorithms' performances reported in Section 5 on the corresponding test sessions, for a total of 6 697 session completion tasks.⁶

5 PERFORMANCE COMPARISON

Table 1 reports the performance of the algorithms in terms of accuracy, novelty, diversity, and popularity bias on the 6 697 test session completions of the 100 randomly sampled users. In terms of accuracy (F_1 and Next-HR), GRU4Rec outperforms the proposed algorithms, as well as the other baselines. In terms of F_1 , GRU4Rec is followed by Temporal UserkNN; this confirms the results from previous work [14, 33], showing that these two algorithms are competitive in the task of sequential recommendation. Interestingly, however, when looking at Next-HR the performance of Temporal UserkNN displays a substantial drop, and is clearly outperformed by MostRecent. This confirms the results reported by Reiter-Haas et al. [40], indicating that recommending the last track of the session is a strong baseline with respect to Next-HR. The fact that approaches based on recurrent neural networks achieve high accuracy of recommendation, and that simply recommending the last track achieves a high Next-HR, indicate that listening sessions tend to display recurring temporal patterns (in the extreme case, repetitions of single tracks). With respect to P-Noveltiy, two of our proposed algorithms achieve the best performances: ACT-R + BPR and Social ACT-R. Social ACT-R achieves the highest values of diversity, indicating that it is able to recommend tracks of various genres for completing a session. It is interesting to observe that both diversity and accuracy of Social ACT-R recommendations are higher compared to ACT-R: The inclusion of collaborative information in Social ACT-R hence increases diversity and F_1 simultaneously. Finally, we observe that simple temporal- or memory-based approaches, i.e., MostRecent and ACT-R, are less biased towards popular tracks. This pattern could be explained by the fact that, since they only consider the listening events of the target user, they do not rely on collaborative data, which is a common source of popularity bias [2, 3, 24, 37].

In summary, we observe that the proposed algorithms – although not outperforming the accuracy of DNN-based algorithms – achieve the highest performance in terms of beyond-accuracy metrics, such as novelty and diversity, are able to provide more accurate novel recommendations (P-Noveltiy), and outperform pure ACT-R models in terms of accuracy and beyond-accuracy metrics.

6 EXPLAINABLE MUSIC RECOMMENDATION

Since the proposed algorithms are based on a well-defined psychological model, their recommendations are intrinsically explainable. This is advantageous for different RS stakeholders, as we discuss in this section.

⁶For reproducibility purposes, we share the code, dataset, details on the dataset handling and splits, hyperparameter optimization, and pretrained instances of BPR and MultVAE required by the algorithms described in Section 3 at <https://github.com/hcaimms/actr>.

Table 1: Performance of the models in the session-completion task. Models are sorted in order of descending F₁ score. All values are averaged over the 6 697 test session completions of the 100 randomly sampled users, as described in Section 4.3. New models are highlighted in blue. Best performances are highlighted in bold, second best are underlined.

	F ₁	Next-HR	Novelty	P-Novelty	Diversity	PopBias
GRU4Rec	0.142	0.198	0.716	0.126	0.929	0.099
Temporal UserkNN	<u>0.122</u>	0.024	0.631	0.146	0.786	0.122
Item BPR	0.114	0.051	0.846	0.130	0.658	0.151
Weighted MultVAE	0.111	0.045	0.554	0.136	<u>0.941</u>	0.194
ACT-R + BPR	0.104	0.037	0.056	0.239	0.923	<u>0.065</u>
Social ACT-R	0.101	0.037	0.056	<u>0.155</u>	0.945	0.066
MostRecent	0.094	<u>0.069</u>	0.000	0.000	0.891	0.060
ACT-R	0.093	0.037	0.000	0.000	0.889	0.060
Weighted MultVAE UserkNN	0.064	0.010	<u>0.831</u>	0.064	0.833	0.176

Figure 1a shows an example of an initial segment of length $l = 6$, of a session of total length $N = 12$. The initial segment consists of five unique tracks, one of them listened to twice. Figure 1b shows the list of $N - l = 6$ tracks with the highest recommendation score according to Social ACT-R. The columns show the relative contribution of the components of Social ACT-R, also reflected as a color gradient. Each component captures a different aspect of relevance for Social ACT-R’s recommendations, and can therefore be translated in a way that can easily be understood by the end user. *Current obsession* corresponds to BLL, which captures the recency (*Current*) and frequency (*obsession*) of interactions with the track. *Current vibes* corresponds to S since this component favors tracks that often occurred together with the last one. *Evergreens* corresponds to V which, with a binary reward, favors tracks that were often listened to by the target user, irrespective of when in the past. Finally, *From similar listeners* corresponds to SC, which reflects collaborative information. Figure 1 hence gives a clear indication of why each song was recommended to the user. The top-5 recommendations all belong to the target user’s current session. The 6th is a track that the target user never listened to, as it is evident from the vanishing ACT-R components. In this particular case, the ACT-R scores all vanish for other elements of the catalog, indicating that in the one-week window used to evaluate the ACT-R scores, only the current session is present. Therefore, in this example ACT-R alone would not allow recommending more than five tracks, while this can be achieved with Social ACT-R. For instance, the track at the top of the recommendation list (*From the Past comes the Storms*) appeared in the target user’s past interactions recently (BLL), often (V and BLL), and in sessions that included the most recent track the user listened to (S). For the last track in the list, the situation is different: this track is not part of the target user’s past interactions; therefore it was recommended since users with a similar listening profile (according to the ACT-R activations) also listened to it. The possibility to investigate the contributions of the different components to the final recommendation score may also provide useful information to the platform providers. In the example provided in Figure 1, for instance, we see that the ACT-R components entirely contribute to the recommendation scores at the top of the list, while SC only becomes relevant once all the tracks of the initial segment of the current session have been recommended. We attribute this to

the fact that the ACT-R activations are nonvanishing for a limited set of tracks (all tracks the target user listened to in the last seven days), while SC does not vanish for a larger set of tracks (all tracks listened to by the k most similar users). Aggregating the individual ACT-R activations and SC as described in Section 3 hence results in a very peaked individual ACT-R distribution over items, and a more spread and almost negligible SC. Therefore, if a platform provider wants to favor the CF component, for instance for providing more novel recommendations, they might consider rank-based aggregation techniques, or consider assigning a higher weight to SC. The explainability of RSs that integrate ACT-R with CF can also be used to discover patterns in recommendations, which can be useful both to platform providers and content producers. To give an example, we analyze how often each of the Social ACT-R components is the *salient* one, i.e., how often the score of this component is larger than the score of the other components. Figure 2 shows the salience of each component, in percentage over all recommendations, and for specific genres. By looking at the salience over all genres, we see that taken together, the ACT-R components are salient for about 90% of all recommendations, with S being the dominant component for more than half of them. Hence, context, i.e., the last track the user listened to, often plays the largest role in selecting which track to recommend. This tendency can change when looking at specific genres. For instance, while for tracks of the genre *non-music* – often corresponding to spoken words – the salience of S is even increased, the situation is inverted for *stage and screen* – e.g., tracks that are part of movie soundtracks. For *stage and screen*, BLL is often the salient component, and together with V is the salient component in above 75% of recommendations. This indicates that for recommendations of *non-music* tracks, the last track listened to is particularly relevant, while for *stage and screen* frequency of occurrence in the past listening events is more important. This information can be further leveraged by the platform providers to weight the relative importance of each component in a genre-specific way, in order to design ACT-R- and content-based RSs, whose recommendations are tailored to genres. Finally, investigating the components’ salience may help content producers to gain insight into the behavior of listeners of specific genres. For instance, *reggae* artists might observe that for this genre V is often the most salient contribution

Session Position	Listened Track
1	The Abyss
2	R.I.P. (Rest in Pain)
3	From the Past Comes the Storms
4	From the Past Comes the Storms
5	To the Wall
6	Escape the Void

(a) Initial segment of length $l = 6$.

Recommended Track	Current obsession (BLL)	Current vibes (S)	Evergreens (V)	From similar listeners (SC)
From the Past Comes the Storms	0.471	0.248	0.281	0.000
Escape to the Void	0.306	0.353	0.341	0.000
To the Wall	0.294	0.359	0.347	0.000
R.I.P. (Rest in Pain)	0.264	0.374	0.362	0.000
The Abyss	0.263	0.375	0.362	0.000
Troops of Doom	0.000	0.000	0.000	1.000

(b) Recommendations for the remaining $N - l = 6$ tracks in the session.

Figure 1: Left: Example of initial segment of length 6 of a target session of total length 12. The column “Session Position” displays the position of the track in the initial segment of the target session. Right: Heatmap of the relative contribution of the used Social ACT-R components to the total recommendation score of each of the 6 recommendations (remaining session length). The more intense the color, the higher the contribution.

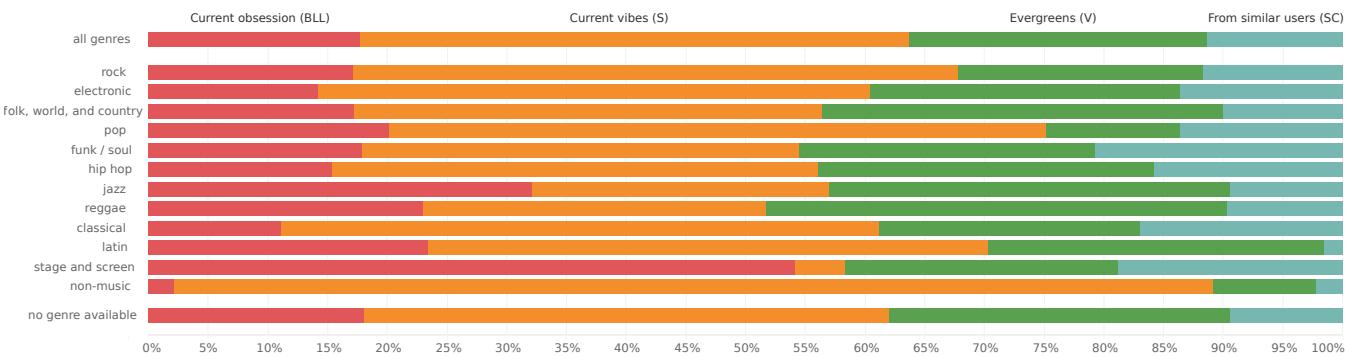


Figure 2: Component salience over all Social ACT-R recommendations and over Social ACT-R recommendations of a specific genre. A component is considered salient if its score is higher than the scores of the other components of the same track. Investigating the components’ salience may help content producers to understand their listeners’ behaviour.

and conclude that their fans have a higher tendency to relisten to the same tracks, irrespective of the last track they listened to.

7 CONCLUSION AND FUTURE WORK

In this work, we proposed four new RS algorithms that integrate the ACT-R cognitive architecture with CF for sequential music recommendation: Social ACT-R, ACT-R + BPR, Weighted MultVAE, and Weighted MultVAE UserkNN. We showed that although the proposed algorithms do not outperform the accuracy of DNN-based recommenders, they achieve the highest performance in terms of beyond-accuracy metrics. In particular, integrating CF with ACT-R in Social ACT-R achieves the highest diversity and simultaneously increases F_1 with respect to ACT-R. More importantly, the proposed algorithms can be used for providing explainable recommendations, which can enhance the users’ engagement with the platform, provide insight to platform providers on the RSs, and to artists on the listening behaviors of their listeners. One of the limitations of this work is that it exclusively considers one perspective from cognitive psychology, i.e., that of the ACT-R model. Additionally, the definition of context for the spreading and partial matching components is given in terms of the last item of the session. While this agrees with the way context is defined in the ACT-R cognitive

architecture, it would be interesting to extend the work with definitions of context that are more common in the RS community, such as location or time of the day. Moreover, we optimized the RSs for achieving the highest F_1 score. Due to the structure of the proposed algorithms, including beyond-accuracy metrics in the optimization process would allow analyzing how each component impacts the different aspects of recommendation. This could be translated to more detailed explanations and be leveraged for the design of a hybrid RS that can be tuned by each user according to their needs. We leave these extensions of our work for future research. Finally, the explainability of the algorithms proposed in this work opens up the possibility to evaluate the quality, user acceptance and understandability of the explanations by means of user studies; we leave this evaluation for future work.

ACKNOWLEDGMENTS

This research was funded in whole, or in part, by the Austrian Science Funds (FWF): P33526 and DFH-23, and by the State of Upper Austria and the Federal Ministry of Education, Science, and Research, through grant LIT-2020-9-SEE-113.

REFERENCES

- [1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. 2020. Multistakeholder recommendation: Survey and research directions. *User Modeling and User-Adapted Interaction* 30 (2020), 127–158.
- [2] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2017. Controlling Popularity Bias in Learning-to-Rank Recommendation. In *Proc. of ACM RecSys* (Como, Italy). 42–46.
- [3] Himan Abdollahpouri, Masoud Mansouri, Robin Burke, Bamshad Mobasher, and Edward Malthouse. 2021. User-Centered Evaluation of Popularity Bias in Recommender Systems. In *Proc. of ACM UMAP* (Utrecht, Netherlands). 119–129.
- [4] Darius Afchar, Alessandro B. Melchiorre, Markus Schedl, Romain Hennequin, Elena V. Epure, and Manuel Moussallam. 2022. Explainability in music recommender systems. *AI Magazine* 43, 2 (2022), 190–208.
- [5] Mehrnaz Amjadi, Seyed Danial Mohseni Taheri, and Theja Tulabandhula. 2021. KATRec: Knowledge Aware aTtentive Sequential Recommendations. In *Proc. of DS*. 305–320.
- [6] John R. Anderson, Daniel Bothell, Michael D. Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. 2004. An integrated theory of the mind. *Psychological review* 111, 4 (2004), 1036.
- [7] Frederick Conrad, Jason Corey, Samantha Goldstein, Joseph Ostrow, and Michael Sadowsky. 2019. Extreme re-listening: Songs people love... and continue to love. *Psychology of Music* 47, 2 (2019), 158–172.
- [8] Gabriel de Souza Pereira Moreira, Sara Rabhi, Jeong Min Lee, Ronay Ak, and Even Oldridge. 2021. Transformers4Rec: Bridging the Gap between NLP and Sequential / Session-Based Recommendation. In *Proc. of ACM RecSys* (Amsterdam, Netherlands). 143–153.
- [9] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. In *Proc. of ACM RecSys*. 101–109.
- [10] Danilo Fum and Andrea Stocco. 2004. Memory, Emotion, and Rationality: An ACT-R interpretation for Gambling Task results. In *Proc. of ICCM* (Pittsburgh, PA, USA). 106–111.
- [11] Diksha Garg, Priyanka Gupta, Pankaj Malhotra, Lovekesh Vig, and Gautam Shroff. 2019. Sequence and Time Aware Neighborhood for Session-based Recommendations: STAN. In *Proc. of ACM SIGIR* (Paris, France). 1069–1072.
- [12] Tajuddeen Rabib Gwadabe and Ying Liu. 2022. IC-GAR: item co-occurrence graph augmented session-based recommendation. *Neural Computing and Applications* 34, 10 (2022), 1–16.
- [13] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations. In *Proc. of ACM RecSys* (Boston, MA, USA). 241–248.
- [14] Dietmar Jannach and Malte Ludewig. 2017. When Recurrent Neural Networks meet the Neighborhood for Session-Based Recommendation. In *Proc. of ACM RecSys* (Como, Italy). 306–310.
- [15] Ion Juvina, Othalia Larue, and Alexander Hough. 2018. Modeling valuation and core affect in a cognitive architecture: The impact of valence and arousal on memory and decision-making. *Cognitive Systems Research* 48 (2018), 4–24.
- [16] Marius Kaminskas and Derek Bridge. 2016. Diversity, Serendipity, Novelty, and Coverage: A Survey and Empirical Analysis of Beyond-Accuracy Objectives in Recommender Systems. *ACM Transactions on Interactive Intelligent Systems* 7, 1 (2016), 42 pages.
- [17] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *Proc. of IEEE ICDM*. 197–206.
- [18] Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2019. Modeling artist preferences of users with different music consumption patterns for fair music recommendations. In *LBR of ISMIR* (Delft, Netherlands).
- [19] Dominik Kowald, Subhash Chandra Pujari, and Elisabeth Lex. 2017. Temporal effects on hashtag reuse in twitter: A cognitive-inspired hashtag recommendation approach. In *Proc. of ACM WWW* (Tacoma, WA USA). 1401–1410.
- [20] Emanuel Lacic, Dominik Kowald, Markus Reiter-Haas, Valentin Slawicek, and Elisabeth Lex. 2017. Beyond Accuracy Optimization: On the Value of Item Embeddings for Student Job Recommendations. *CoRR* (2017).
- [21] Emanuel Lacic, Dominik Kowald, Paul Christian Seitlinger, Christoph Trattner, and Denis Parra. 2014. Recommending Items in Social Tagging Systems Using Tag and Time Information. In *In Proceedings of the 1st Social Personalization Workshop co-located with the 25th ACM Conference on Hypertext and Social Media*. Association of Computing Machinery, 4–9.
- [22] Emanuel Lacic, Markus Reiter-Haas, Tomislav Duricic, Valentin Slawicek, and Elisabeth Lex. 2019. Should we embed? A study on the online performance of utilizing embeddings for real-time job recommendations. In *Proc. of ACM RecSys* (Copenhagen, Denmark). 496–500.
- [23] Oleg Lesota, Stefan Brandl, Matthias Wenzel, Alessandro B. Melchiorre, Elisabeth Lex, Navid Rekabsaz, and Markus Schedl. 2022. Exploring Cross-group Discrepancies in Calibrated Popularity for Accuracy/Fairness Trade-off Optimization. In *Proc. of MORS RecSys* (Seattle, WA, USA).
- [24] Oleg Lesota, Alessandro Melchiorre, Navid Rekabsaz, Stefan Brandl, Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2021. Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected?. In *Proc. of ACM RecSys* (Amsterdam, Netherlands). 601–606.
- [25] Elisabeth Lex, Dominik Kowald, and Markus Schedl. 2020. Modeling popularity and temporal drift of music genre preferences. *Transactions of the International Society for Music Information Retrieval* 3, 1 (2020).
- [26] Elisabeth Lex, Dominik Kowald, Paul Seitlinger, Thi Ngoc Trang Tran, Alexander Felfernig, Markus Schedl, et al. 2021. Psychology-informed recommender systems. *Foundations and Trends in Information Retrieval* 15, 2 (2021), 134–242.
- [27] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural Attentive Session-based Recommendation. In *Proc. of ACM CIKM*. ACM, 1419–1428.
- [28] Taoying Li, Linlin Jin, Zebin Wu, and Yan Chen. 2019. Combined Recommendation Algorithm Based on Improved Similarity and Forgetting Curve. *MDPI Information* 10, 4 (2019).
- [29] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational Autoencoders for Collaborative Filtering. In *Proc. of ACM WWW* (Lyon, France). 689–698.
- [30] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 2018. STAMP: Short-Term Attention/Memory Priority Model for Session-based Recommendation. In *Proc. of ACM SIGKDD*. 1831–1839.
- [31] Malte Ludewig and Dietmar Jannach. 2018. Evaluation of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction* 28 (2018), 331–390.
- [32] Malte Ludewig, Iman Kamekhosh, Nick Landia, and Dietmar Jannach. 2018. Effective Nearest-Neighbor Music Recommendations. In *Proc. of ACM RecSys Challenge* (Vancouver, BC, Canada). Article 3, 6 pages.
- [33] Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. 2019. Performance Comparison of Neural and Non-Neural Approaches to Session-Based Recommendation. In *Proc. of ACM RecSys* (Copenhagen, Denmark). 462–466.
- [34] Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. 2021. Empirical analysis of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction* 31, 1, 149–181.
- [35] Alessandro B. Melchiorre, Navid Rekabsaz, Christian Ganhoer, and Markus Schedl. 2022. ProtoMF: Prototype-based Matrix Factorization for Effective and Explainable Recommendations. In *Proc. of ACM RecSys* (Seattle, WA, USA). 246–256.
- [36] Alessandro B. Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. 2021. Investigating gender fairness of recommendation algorithms in the music domain. *Information Processing & Management* 58, 5 (2021), 102666.
- [37] Yoon-Joo Park and Alexander Tuzhilin. 2008. The Long Tail of Recommender Systems and How to Leverage It. In *Proc. of ACM RecSys* (Lausanne, Switzerland). New York, NY, USA, 11–18.
- [38] Andreas Peintner, Marta Moscati, Emilia Parada-Cabaleiro, Markus Schedl, and Eva Zangerle. 2022. Unsupervised Graph Embeddings for Session-based Recommendation with Item Features. In *Proc. of MORS RecSys*.
- [39] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. 2018. Sequence-aware recommender systems. *ACM Computing Surveys (CSUR)* 51, 4 (2018), 1–36.
- [40] Markus Reiter-Haas, Emilia Parada-Cabaleiro, Markus Schedl, Elham Motamed, Marko Tkalcic, and Elisabeth Lex. 2021. Predicting Music Relistening Behavior Using the ACT-R Framework. In *Proc. of ACM RecSys* (Amsterdam, Netherlands). 702–707.
- [41] Lei Ren. 2015. A Time-Enhanced Collaborative Filtering Approach. In *Proc. of NCIT* (Qingdao, China). 7–10.
- [42] Pengjie Ren, Zhumin Chen, Jing Li, Zhaochun Ren, Jun Ma, and Maarten de Rijke. 2019. RepeatNet: A Repeat Aware Neural Recommendation Machine for Session-Based Recommendation. In *Proc. of AAAI* (Honolulu, Hawaii, USA). Article 590, 8 pages.
- [43] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proc. of UAI* (Montreal, Quebec, Canada). 452–461.
- [44] Frank E. Ritter, Farnaz Tehranchi, and Jacob D. Orry. 2018. ACT-R: A cognitive architecture for modeling cognition. *WIREs Cognitive Science* 10, 3 (2018), e1488.
- [45] Markus Schedl, Stefan Brandl, Oleg Lesota, Emilia Parada-Cabaleiro, David Penz, and Navid Rekabsaz. 2022. LFM-2b: A Dataset of Enriched Music Listening Events for Recommender Systems Research and Fairness Analysis. In *Proc. of ACM CHIIR* (Regensburg, Germany). 337–341.
- [46] Markus Schedl, Emilia Gómez, and Julián Urbano. 2014. Music Information Retrieval: Recent Developments and Applications. *Foundations and Trends in Information Retrieval* 8, 2–3 (2014), 127–261.
- [47] Markus Schedl, Peter Knees, Brian McFee, and Dmitry Bogdanov. 2022. *Music Recommendation Systems: Techniques, Use Cases, and Challenges*. Springer US, New York, NY, 927–972.
- [48] Markus Schedl, Hamed Zamani, Ching-Wei Chen, Yashar Deldjoo, and Mehdi Elahi. 2018. Current Challenges and Visions in Music Recommender Systems Research. *International Journal of Multimedia Information Retrieval* 7, 2 (2018),

95–116.

- [49] Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved Recurrent Neural Networks for Session-Based Recommendations. In *Proc. of DLRS* (Boston MA USA). 17–22.
- [50] Baocheng Wang and Wentao Cai. 2020. Knowledge-Enhanced Graph Neural Networks for Sequential Recommendation. *MDPI Information* 11, 8 (2020), 388.
- [51] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-Based Recommendation with Graph Neural Networks. *Proc. of AAAI* 33, 01, 346–353.
- [52] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. 2019. Graph Contextualized Self-Attention Network for Session-based Recommendation. In *Proc. of IJCAI*. 3940–3946.
- [53] Liangliang Zhao, Jiajin Huang, and Ning Zhong. 2014. A context-aware recommender system with a cognition inspired model. In *Proc. of RSKT* (Shanghai, China). 613–622.

B.2 Privacy and Limited Preference Information in Recommender Systems

P6 Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence (2018)

Privacy and Limited Preference Information in Recommender Systems

[P6] Duricic, T., Lacic, E., **Kowald, D.**, Lex, E. (2018). Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys'2018)*, pp. 446–450.

DOI: <https://doi.org/10.1145/3240323.3240404>



Trust-Based Collaborative Filtering: Tackling the Cold Start Problem Using Regular Equivalence

Tomislav Duricic

Know-Center GmbH & Graz University of Technology
Graz, Austria
tduricic@know-center.at

Dominik Kowald

Know-Center GmbH
Graz, Austria
dkowald@know-center.at

Emanuel Lacic

Know-Center GmbH
Graz, Austria
elacic@know-center.at

Elisabeth Lex

Know-Center GmbH & Graz University of Technology
Graz, Austria
elisabeth.lex@tugraz.at

ABSTRACT

User-based Collaborative Filtering (CF) is one of the most popular approaches to create recommender systems. This approach is based on finding the most relevant k users from whose rating history we can extract items to recommend. CF, however, suffers from data sparsity and the cold-start problem since users often rate only a small fraction of available items. One solution is to incorporate additional information into the recommendation process such as explicit trust scores that are assigned by users to others or implicit trust relationships that result from social connections between users. Such relationships typically form a very sparse trust network, which can be utilized to generate recommendations for users based on people they trust. In our work, we explore the use of regular equivalence applied to a trust network to generate a similarity matrix that is used to select the k -nearest neighbors for recommending items. We evaluate our approach on Epinions and we find that we can outperform related methods for tackling cold-start users in terms of recommendation accuracy.

KEYWORDS

Trust; Recommender Systems; Collaborative Filtering; Cold-start; Network Science; Regular Equivalence; Katz similarity

1 INTRODUCTION

Ever since their introduction, user-based Collaborative Filtering (CF) approaches have been one of the most widely adopted and studied algorithms in the recommender systems literature [21]. CF is based on the intuition that those users, who have shown similar item rating behavior in the past, will likely give similar ratings to items in the future. Typically, CF comprises of three steps: first, we retrieve the k -nearest neighbors to the target user for whom the recommendations are generated. Second, we employ the ratings from these k neighbors to determine items, which were rated highly

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

RecSys '18, Vancouver, BC, Canada

© 2018 ACM. 978-1-4503-5901-6/18/10...\$15.00

DOI: 10.1145/3240323.3240404

by them but have not yet been rated by the target user. Third, these items are weighted or ranked by applying an appropriate algorithm.

In practice, each user's ratings are stored in a rating vector. These rating vectors are then used to calculate the correlation between the target user's vector and rating vectors of the rest of the users. The higher the correlation between the rating vectors of two users, the higher their similarity. This can be assessed, e.g., via the Pearson's correlation coefficient, Cosine similarity, Jaccard index or Mean Squared Difference (MSD) [16, 20]. However, such an approach to neighbor selection suffers from a cold-start user problem. This term refers to novel users which have rated a small number of items or have not yet rated any items at all [14, 22]. This means that we cannot use their rating vectors for finding similar users based on the pairwise vector correlation measure.

Apart from popularity-based or location-based approaches [11, 13, 19], trust-based CF methods have been suggested to mitigate cold-start user problems. Their basis are trust statements expressed on platforms such as, e.g., Epinions [17]. Trust statements can either be expressed explicitly by, for example, assigning trust scores or implicitly by engaging in social connections with trusted users. Based on such trust statements, trust networks can be created with the aim to generate recommendations for users based on people they trust [15]. Since trust networks are often also sparse, a particular property of trust, namely transitivity [2], can be exploited to propagate trust in the network. In this way, new connections are established between users, who are not directly connected, but are connected via intermediary users. Previous work in this respect proposed to perform a modified breadth first search in the trust network to compute a prediction. For example, TidalTrust [4] aggregates and weights the trust values between direct neighbors of two users. MoleTrust [17] works in a similar fashion, but does a backward exploration while considering all users up to a pre-defined maximum depth. In order to efficiently avoid the impact of noisy data while still considering enough ratings, the authors of [10] proposed TrustWalker. They combined trust-based and item-based recommendations, where a random walk model is utilized to compute the confidence in the predictions.

Present work. In this work, we focus on the first step of CF, i.e., finding the k -nearest neighbors. For this purpose, we explore the use of a similarity measure from network science referred to as "Katz similarity" (KS) by the author of [18]. Although Katz himself

never discussed it, KS captures regular equivalence of nodes in a network and can be applied in many different settings [6, 7]. As such, in this work, we explore how to use KS in a trust-based CF approach.

Firstly, we utilize the trust connections to create an adjacency matrix where each entry represents a directed trust link between two users. Secondly, we apply the KS measure on the created trust adjacency matrix. More specifically, we calculate the pairwise similarities between users by using the iterative approach on calculating KS. The iterative approach does not only allow us to calculate the similarity between two nodes in the network, but additionally provides the possibility to choose the maximum used path length in doing so. This approach effectively gives us the ability to decide how far do we want to propagate trust in the network. Lastly, we use the resulting similarity matrix and apply various normalization techniques in order to get a better distribution of similarity values and better evaluation results in return. We evaluate these approaches on the Epinions dataset.

Contributions and findings. The contributions of this work are three-fold: (i) we explore the application of KS measure in the neighbor selection step of the trust-based CF approach for cold-start users, (ii) we evaluate different normalization techniques on the resulting similarity matrix to achieve better recommendation accuracy, and (iii) we introduce an adapted KS measure that gives higher similarity values to node pairs with path lengths of 2. In the trust-based CF setting, this means that propagated trust connections are given a higher importance than by using the standard KS measure.

Taken together, this study may help researchers to get an insight on how to apply KS on trust networks in combination with different normalization techniques to address the cold-start user problem in CF-based recommender systems. Moreover, we show that our approach for boosting the propagated trust values can result in increasing the impact of newly created trust connections on recommendation accuracy.

2 APPROACH

Our approach utilizes Katz similarity, which is a measure of regular equivalence, i.e., a measure of the extent to which two nodes share the same neighbors but also the extent to which their neighbors are similar. As described in [3], two nodes may have few or no neighbors in common, but they may still be similar in an indirect, global way. The idea behind KS is that paths of any length are contributing to the value of similarity between two nodes in the network, with shorter paths having a stronger impact. KS can be mathematically expressed in a matrix form as follows:

$$\sigma = \sum_{l=0}^{\infty} (\alpha A)^l = (I - \alpha A)^{-1} \quad (1)$$

where σ represents the similarity matrix and each value $\sigma_{i,j}$ is a similarity value between nodes i and j , A represents the adjacency matrix of the network, I is the identity matrix which is necessary to make sure that each node is similar to itself, α is the attenuation factor which weights the contribution of a path of length l . In our trust-based setting, the adjacency matrix A is asymmetric and it

represents an unweighted directed trust network, in which each node corresponds to a single user and each link represents a trust statement issued by one user to another:

$$A_{i,j} = \begin{cases} 1, & \text{if user } j \text{ expressed a trust statement to user } i \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

This also makes the similarity matrix σ asymmetric, which means that $\sigma_{i,j}$ does not have to be equal to $\sigma_{j,i}$, which is of advantage because in this way, the asymmetric property of trust is preserved. Furthermore, one important thing to note is that for (1) to converge, the attenuation factor has to satisfy the following condition:

$$\alpha < \frac{1}{\lambda_A} \quad (3)$$

where λ_A is the largest eigenvalue of A . The largest eigenvalue for the Epinions trust network (see Section 3) is 120.54, hence α needs to be less than 0.0083 and we set it to 0.008 throughout all of our experiments.¹ Since calculating the matrix inverse is computationally expensive, we can evaluate the above summation expression starting from $l = 0$ for a fixed maximum l (i.e., l_{max}) and get the following:

$$\begin{aligned} \sigma^{(0)} &= 0 \\ \sigma^{(1)} &= I \\ \sigma^{(2)} &= \alpha A + I \\ \sigma^{(3)} &= \alpha^2 A^2 + \alpha A + I \\ &\dots \\ \sigma^{(l_{max}+1)} &= \sum_{l=0}^{l_{max}} (\alpha A)^l \end{aligned} \quad (4)$$

Step 1: Setting l_{max} . By using this approach and setting l_{max} to a positive integer value, we can define how far down the network do we want to propagate similarity or in this case, trust. In the conducted experiments, we used values 1 and 2 as l_{max} , which means that we either have not propagated similarities through the network at all or that we propagated them through the network using a maximum path length of 2.

Step 2: Degree normalization. As described in [18], σ as defined in (1), tends to give high similarity to nodes that have a high degree. In some cases this might be desirable but if we want to get rid of this bias, we could apply a degree normalization on σ , which would give higher similarity values to pairs of nodes that, independently of their degrees, are similar, while lower values would correspond to pairs of nodes that are dissimilar. Mathematically, for a given l_{max} , this step can be written as follows:

$$\sigma_{Dnorm}^{(l_{max}+1)} = D^{-1} \left(\sum_{l=0}^{l_{max}} (\alpha A)^l \right) D^{-1} \quad (5)$$

¹Although we used the iterative approach to calculate KS where l_{max} was set to a small integer value and α could have been set to any value between 0 and 1, we investigate the impact of α when the condition in Eq. 3 is also satisfied.

where \mathbf{D} represents a degree matrix of a network. In the conducted experiments, we evaluated approaches with an in-degree normalization, a combined-degree normalization and without a degree normalization.²

Step 3: Row normalization. After applying degree normalization, we found that all of the values in the degree normalized similarity matrix are very close to 0, including the maximum value. Therefore, we introduced an additional step where we individually scale rows of the final resulting matrix using one of the three vector norms: l_1 , l_2 or max .³

Step 4: Boosting propagated similarities. As already mentioned, the attenuation factor α is used to decrease similarity the further it gets propagated in the network. Since we set the α to 0.008, similarity decays fast with each propagation step. Therefore, propagated similarity values become much smaller already in the first propagation step, i.e., for $l = 2$. This would mean that trust connections created through propagation in comparison with direct trust connections have an almost insignificant impact on the resulting recommendations, and therefore, this additional boosting step would increase the impact of propagated similarities with respect to the recommendation accuracy.

Largest value for l_{max} in the conducted experiments was set to 2. This could be interpreted as using user's neighbors and their neighbors for generating item recommendations. One of the contributions of this paper was to increase the impact of propagated trust values generated with KS for $l_{max} = 2$. Our proposed approach for doing so consists of the following four steps: (i) calculate $\sigma^{(3)}$ as described above using trust network as \mathbf{A} , (ii) create a new similarity matrix $\hat{\sigma}$ such that:

$$\hat{\sigma}_{i,j} = \begin{cases} \sigma_{i,j}^{(3)}, & \text{if } A_{i,j} = 0 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

(iii) create $\hat{\sigma}_{norm}$ matrix by individually scaling rows of $\hat{\sigma}$ using l_1 , l_2 or max vector norm and lastly, (iv) create a similarity matrix σ_{boost} such that:

$$\sigma_{boost} = \mathbf{A} + \hat{\sigma}_{norm} \quad (7)$$

With this approach, we achieve that each entry in σ_{boost} has a similarity value of 1 between pairs of nodes for which there exists an explicit trust connection in \mathbf{A} and for pairs of nodes for which the similarity has been calculated through propagation, the similarity values are not exclusively small values close to zero increasing their impact on the resulting recommendations.

Recommendation strategy. As already outlined in Section 1, in this work, we focus on user-based CF. We first create a similarity matrix using the above mentioned four steps: (i) calculate σ using Eq. (4) with $l_{max} \in \{1, 2\}$, (ii) normalize the similarity matrix using in-degree or combined-degree normalization, (iii) normalize similarity matrix rows using l_1 , l_2 or max vector norm, and (iv) apply boosting of propagated similarities. Steps (ii), (iii) and (iv)

²Combined-degree matrix is a diagonal matrix where each value on the diagonal corresponds to the sum of in-degree and out-degree of a particular node.

³For example, by utilizing the scikit-learn library in Python: <http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.normalize.html>.

are optional and can be skipped. Utilizing the created trust-based similarity matrix, we first find the k -nearest similar users and afterwards recommend the items of those users as a ranked list of top- n items to the target user. According to the literature, the maximum number of nearest neighbors should be a value between 20 and 60 [8], we used 60 in all of our experiments. The final ranking of the items to recommend is calculated by summing up the similarities of neighboring users as done in [11, 23].

3 EXPERIMENTAL SETUP

Dataset. To evaluate the performance of our trust-based CF approaches for cold-start users, the well-known *Epinions* dataset has been used [17]. This dataset was crawled from the consumer reviewing platform Epinions.com. Here, registered users can rate items available on the Epinions platform on a scale of 1 – 5. Additionally, users can issue trust statements to other users on the platform, i.e., they can express how much they trust other users. In this dataset, there are only positive values for trust statements, meaning there are no negative trust statements (i.e., distrust).

Taken together, there is a total number of 49,290 users in our dataset, which rated 139,738 different items with 664,824 ratings. Moreover, users have issued a total number of 487,181 trust connections. We utilized the trust connections issued by the users to create an unweighted trust network, in which each node represents a user and each directed link represents a trust statement expressed by one user to another. The resulting trust network provides a graph density value of 0.0002, making the trust network adjacency matrix very sparse.

Baseline algorithms. We compare our proposed approach to three baselines algorithms from the literature, which were shown to be useful methods in cold-start settings:

MP. MostPopular is a classic approach in recommender systems, which recommends the most frequently used items in the dataset to every user. Thus, it can be also applied in a cold-start setting.

Trust_{exp}. This naive trust-based approach uses explicit trust values in order to create the neighborhood of a user. Basically, adjacency matrix \mathbf{A} created from a trust network is used as a similarity matrix which does not allow for ranking of similar users because similarity values are binary, i.e., either 0 or 1.

Trust_{jac}. This is a trust-based approach using Jaccard coefficient on explicit trust values and was also used by the authors of [1]. The idea behind this approach is that two users are more similar the more trusted users they have in common. Jaccard coefficient is a statistic used to measure the similarity and diversity of sample sets and it can be written as:

$$J(\mathbf{A}_{*,a}, \mathbf{A}_{*,b}) = \frac{|\mathbf{A}_{*,a} \cap \mathbf{A}_{*,b}|}{|\mathbf{A}_{*,a} \cup \mathbf{A}_{*,b}|} \quad (8)$$

where $J(\mathbf{A}_{*,a}, \mathbf{A}_{*,b})$ is used to calculate similarity between users a and b , $\mathbf{A}_{*,a}$ corresponds to explicit values given to other users in the trust network by user a and the same applies to $\mathbf{A}_{*,b}$ for user b .

Evaluation method and metrics. In order to compare our proposed approach to these baseline algorithms in a cold-start setting, we extracted all users with no more than 10 rated items from the dataset. This resulted in 25,393 users, for which we put all of their

Approach	l_{max}	Degree norm.	Row norm.	Boost	nDCG	R	P
<i>Trust_{exp}</i>					.0224	.0296	.0110
<i>Trust_{jac}</i>					.0176	.0219	.0087
<i>MP</i>					.0134	.0202	.0070
<i>KSPCMB</i>	2	Combined	Max	Yes	.0303	.0425	.0117
<i>KSPCMN</i>	2	Combined	Max	No	.0295	.0422	.0113
<i>KSPCL₁B</i>	2	Combined	L1	Yes	.0273	.0358	.0106
<i>KSPNL₂B</i>	2	No degree	L2	Yes	.0257	.0340	.0106
<i>KSNCMN</i>	1	Combined	Max	No	.0213	.0289	.0106
<i>KSNINN</i>	1	In degree	N/A	No	.0161	.0243	.0087
<i>KSPNNN</i>	2	No degree	N/A	No	.0036	.0057	.0020

Table 1: Evaluation results for $n = 10$. The reported subset of the 33 evaluated KS-based approaches are additionally labeled for an easier result comparison between different step combinations (i.e., columns 2 to 5).

rated items into the test set. To finally quantify the performance of our evaluated algorithms, we used the well-established accuracy metrics *nDCG*, *Precision* and *Recall* for $n = 1 - 10$ recommended items [9, 24].

4 RESULTS

In our study, we evaluated 33 approaches for all possible step combinations when creating the similarity matrix (i.e., as defined in Section 2). However, for the sake of space, in Table 1, we only report the results for a subset of these approaches that provide the most insightful findings. All of the evaluation results are reported for $n = 10$, i.e., for 10 recommended items. As it can be seen in Table 1, the best performing approach in terms of all accuracy measures was *KSPCMB*, where we used trust propagation ($l_{max} = 2$) with combined degree normalization, row normalization with *max* norm as well as boosting of the propagated similarity values.

One interesting finding was that if similarity propagation was not used, i.e., l_{max} was set to 1, better results were achieved if degree and row normalization were not applied (i.e., basically the *Trust_{exp}* baseline). However, if l_{max} was set to 2, we noticed result improvements in almost all of the cases except when no row normalization was applied, e.g., in the case of *KSPNNN*.

Additionally, similarity propagation with $l_{max} = 2$ increased the similarity matrix density from 0.0002 to 0.008. It turned out that row normalization was a very important step in using KS with similarity propagation for neighbor selection. Another important finding was that the combined-degree normalization provided better results than in-degree normalization in most of the cases. Also, with respect to row normalization, *max* norm provided better results than *l1* and *l2* norms in most of the cases. Lastly, with degree normalization and row normalization unchanged, boosting of propagated similarities often provided better results.

Finally, in Figure 1, we show the performance of all approaches listed in Table 1 in form of Recall-Precision plots for different number of recommended items (i.e., $n = 1 - 10$). The results clearly show that the best performing algorithm (i.e., *KSPCMB*) again outperforms all of the other approaches also for a smaller number of recommended items (i.e., for $n < 10$).

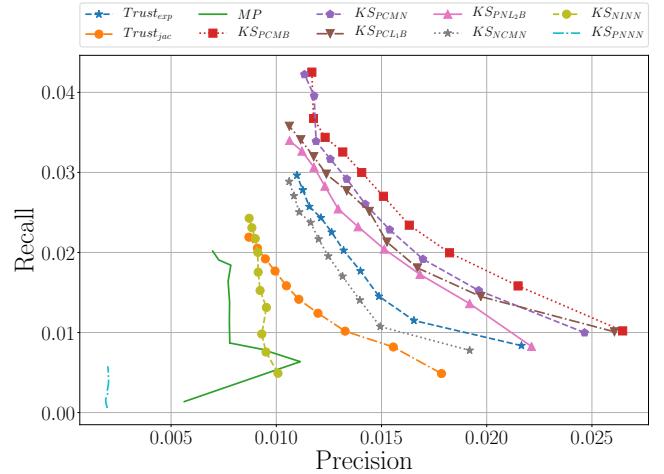


Figure 1: Recall-Precision plots of the described approaches for $n = 1 - 10$ recommended items. Again, we can observe that the approach labeled as *KSPCMB* outperforms all three baselines as well as the other KS-based approaches.

5 CONCLUSION & FUTURE WORK

In this paper, we explored the use of Katz similarity (KS), a similarity measure of regular equivalence in networks, for selecting k -nearest neighbors in a Collaborative Filtering (CF) algorithm for cold-start users. We used an iterative approach for calculating KS since it provides the ability to restrict the length of paths in the network used for similarity calculation. We found that KS can be a very useful measure for neighbor selection if it is used with degree-normalization and row normalization, especially when using similarity propagation. When these techniques are properly combined with KS, we managed to outperform related approaches for tackling the cold-start problem. Our results also indicate that trust propagation is a very important feature when using trust networks in a CF setting as well as that KS is a useful technique for efficiently propagating trust in a network. Summed up, our study may help researchers to get an insight on how to apply KS on trust networks in combination with different normalization techniques to address the cold-start user problem in recommender systems.

One limitation of this study was that we only evaluated our approaches using recommender accuracy, although optimizing on non-accuracy measures has been closely tied to user satisfaction [12, 25]. As such, in the future we plan to investigate the impact of trust-based networks on beyond accuracy metrics such as novelty, diversity and coverage. Additionally, we would like to evaluate our approach not only on cold-start users, but rather to run the experiments on the complete dataset. We would also like to run additional experiments using different values for α and l_{max} . Moreover, we also plan to explore the use of recently popularized node embeddings (e.g., Node2Vec [5]) for trust networks to further improve our results. And finally, we plan to conduct a more extensive evaluation to see how our method compares with other popular approaches which support trust propagation [4, 10, 17].

Acknowledgments. This work was supported by the Know-Center (Austrian COMET program) and the AFEL project (GA: 687916).

REFERENCES

- [1] P. H. Chia and G. Pitsilis. Exploring the use of explicit trust links for filtering recommenders: a study on opinions. com. *Journal of Information Processing*, 19:332–344, 2011.
- [2] S. Fazeli, B. Loni, A. Bellogin, H. Drachsler, and P. Sloep. Implicit vs. explicit trust in social matrix factorization. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, pages 317–320, New York, NY, USA, 2014. ACM.
- [3] M. Franceschet. Global similarity. Available at <https://www.sci.unich.it/~francesc/teaching/network/globalsimilarity.html>. Accessed: 2018-05-05.
- [4] J. A. Golbeck. *Computing and applying trust in web-based social networks*. PhD thesis, 2005.
- [5] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864. ACM, 2016.
- [6] I. Hasani-Mavriqi, D. Kowald, D. Helic, and E. Lex. Consensus dynamics in online collaboration systems. *Computational social networks*, 5(1):2, 2018.
- [7] D. Helic. Regular equivalence in informed network search. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*, pages 1088–1093. IEEE, 2014.
- [8] J. Herlocker, J. A. Konstan, and J. Riedl. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 5(4):287–310, 2002.
- [9] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22, 2004.
- [10] M. Jamali and M. Ester. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 397–406. ACM, 2009.
- [11] E. Lacic, D. Kowald, L. Eberhard, C. Trattner, D. Parra, and L. B. Marinho. Utilizing online social networks and location-based data to recommend products and categories in online marketplaces. In *Mining, Modeling, and Recommending 'Things' in Social Media*, pages 96–115. Springer, 2015.
- [12] E. Lacic, D. Kowald, M. Reiter-Haas, V. Slawicek, and E. Lex. Beyond accuracy optimization: On the value of item embeddings for student job recommendations. *arXiv preprint arXiv:1711.07762*, 2017.
- [13] E. Lacic, D. Kowald, M. Traub, G. Luzhnica, J. Simon, and E. Lex. Tackling cold-start users in recommender systems with indoor positioning systems. In *RecSys Posters*, 2015.
- [14] X. N. Lam, T. Vu, T. D. Le, and A. D. Duong. Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, pages 208–211. ACM, 2008.
- [15] N. Lathia, S. Hailes, and L. Capra. Trust-based collaborative filtering. In *IFIP international conference on trust management*, pages 119–134. Springer, 2008.
- [16] H. Liu, Z. Hu, A. Mian, H. Tian, and X. Zhu. A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, 56:156–166, 2014.
- [17] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24. ACM, 2007.
- [18] M. Newman. *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA, 2010.
- [19] S.-T. Park and W. Chu. Pairwise preference regression for cold-start recommendation. In *Proceedings of the third ACM conference on Recommender systems*, pages 21–28. ACM, 2009.
- [20] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.
- [21] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.
- [22] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260. ACM, 2002.
- [23] P. Seitlinger, D. Kowald, S. Kopeinik, I. Hasani-Mavriqi, E. Lex, and T. Ley. Attention please! a hybrid resource recommender mimicking attention-interpretation dynamics. In *Proceedings of the 24th International Conference on World Wide Web*, pages 339–345. ACM, 2015.
- [24] B. Smyth and P. McClave. Similarity vs. diversity. In *Proceedings of the 4th International Conference on Case-Based Reasoning: Case-Based Reasoning Research and Development*, ICCBR '01. Springer-Verlag, 2001.
- [25] Y. C. Zhang, D. Ó Séaghdha, D. Quercia, and T. Jambor. Auralist: introducing serendipity into music recommendation. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 13–22. ACM, 2012.

**P7 Using Autoencoders for Session-based Job Recommendations
(2020)**

Privacy and Limited Preference Information in Recommender Systems

[P7] Lacic, E., Reiter-Haas, M., **Kowald, D.**, Dareddy, M., Cho, J., Lex, E. (2020). Using Autoencoders for Session-based Job Recommendations. *User Modeling and User-Adapted Interaction*, 30, pp. 617–658.

DOI: <https://doi.org/10.1007/s11257-020-09269-1>



Using autoencoders for session-based job recommendations

Emanuel Lacic¹ · Markus Reiter-Haas² · Dominik Kowald¹ ·
Manoj Reddy Daredy³ · Junghoo Cho³ · Elisabeth Lex⁴

Received: 15 April 2019 / Accepted in revised form: 1 June 2020 / Published online: 1 July 2020
© The Author(s) 2020

Abstract

In this work, we address the problem of providing job recommendations in an online session setting, in which we do not have full user histories. We propose a recommendation approach, which uses different autoencoder architectures to encode sessions from the job domain. The inferred latent session representations are then used in a k-nearest neighbor manner to recommend jobs within a session. We evaluate our approach on three datasets, (1) a proprietary dataset we gathered from the Austrian student job portal Studio Jobs, (2) a dataset released by XING after the RecSys 2017 Challenge and (3) anonymized job applications released by CareerBuilder in 2012. Our results show that autoencoders provide relevant job recommendations as well as maintain a high coverage and, at the same time, can outperform state-of-the-art session-based recommendation techniques in terms of system-based and session-based novelty.

Keywords Job recommendations · Session-based recommendation · Autoencoders · Session embeddings · Accuracy · Novelty

1 Introduction

People increasingly use business-oriented social networks such as LinkedIn¹ or XING² to attract recruiters and to look for jobs (Kenthapadi et al. 2017). Users of such networks make an effort to create personal profiles that best describe their skills, interests, and previous work experience. Even with such carefully structured content, it remains a non-trivial task to find relevant jobs (Abel 2015). As a consequence, the field of job recommender systems has gained much traction in academia

¹ <http://linkedin.com>.

² <http://xing.com>.

✉ Elisabeth Lex
elisabeth.lex@tugraz.at

Extended author information available on the last page of the article

and the industry (Lacic et al. 2019; Siting et al. 2012). The main challenge that job recommender systems tackle is to retrieve a list of jobs for a user based on her preferences or to generate a list of potential candidates for recruiters based on the job's requirements (Hong et al. 2013).

Besides, most online job portals offer the option to browse the available jobs anonymously in order to attract users to the portal. As a consequence, the only data a recommender system can exploit are anonymous user interactions with job postings during a session. In other words, the problem of recommending jobs is a session-based recommendation problem (Jannach and Ludewig 2017). That is, the aim is to recommend the next relevant job in an anonymous session.

In our ongoing work with the Austrian start-up Studio,³ we have started to address the problem of recommending jobs in a session-based environment. In their student job portal Studio Jobs,⁴ we have observed an increasing volume of anonymous user sessions that look for new jobs.⁵ For example, over the past six months, anonymous job-related browsing has doubled from approximately 30,000 to 60,000 job interactions. Therefore, in this paper, we address the problem of recommending jobs in a session-based environment.

Recently, neural networks have gained attention in the context of session-based recommender systems (e.g., Hidasi et al. 2015; Li et al. 2017; Lin et al. 2018; Wu et al. 2018, 2019; Yuan et al. 2019). The idea is to extract latent information about a user's preferences from anonymous, short-lived sessions. For example, autoencoders (Kramer 1991) are neural networks designed to learn meaningful representations, i.e., embeddings, and to reduce the dimensionality of input data. Example applications are data compression (Theis et al. 2017), clustering and dimensionality reduction (Makhzani et al. 2015) as well as recommender systems, where they have been used to find latent similarities between users and items and to predict user preferences (Sedhain et al. 2015; Strub et al. 2016).

Their ability to preserve the most relevant features, while reducing dimensionality, inspired our idea to explore the use of autoencoders to infer latent session representations in the form of embeddings and to use these embeddings to generate recommendations in a k-nearest-neighbor manner. To that end, in this paper, we introduce a recommendation approach, which employs different autoencoder architectures, (1) a classic autoencoder (Kramer 1991), (2) a denoising autoencoder (Vincent et al. 2008) and (3) a variational autoencoder (Jordan et al. 1999), to learn embeddings of job browsing sessions. The inferred latent session representations are then used in a k-nearest neighbor manner to recommend jobs within a session. Besides, we use two types of input data to train and test our approach, i.e., interaction data from sessions and content features of job postings, for which interactions took place during a session. We assess the performance of our approach in the form

³ <https://studio.co>.

⁴ The jobs platform in Studio, which is the predecessor of the Talto career platform (<https://talto.com>)

⁵ We observe this trend independent from the changes in authenticated sessions, which fluctuate heavily over the year. The cause of this trend is that both the total number of sessions and the average ratio of anonymous sessions to authenticated sessions are growing.

of offline evaluations on three datasets from the job domain: firstly, a dataset collected from the Austrian online student job portal Studio Jobs; secondly, the job dataset that was provided by XING after the RecSys Challenge 2017 (Abel et al. 2017); and finally, a dataset from a Kaggle competition on job recommendation sponsored by CareerBuilder. Our approach is compared to the state-of-the-art session-based recommender approaches (Hidasi and Karatzoglou 2018; Hidasi et al. 2015; Jannach and Ludewig 2017; Ludewig and Jannach 2018; Rendle et al. 2009) not only with respect to accuracy but also in terms of system-based and session-based novelty as well as coverage (Zhang et al. 2012). This is grounded in the growing awareness that factors other than accuracy contribute to the quality of recommendations (Herlocker et al. 2004; McNee et al. 2006). Moreover, novelty is especially an important metric for the job domain since applying to popular jobs may decrease a user's satisfaction due to high competition and less chance of getting hired (see e.g., Kenthapadi et al. 2017).

Contributions and findings The main contributions of this paper and the corresponding findings are as follows:

- We present a recommendation approach, which uses different autoencoder architectures to encode sessions from the job domain. We use the inferred latent session representations in a k-nearest neighbor manner to recommend jobs within a session.
- We compare our approach to methods from recent work (Hidasi and Karatzoglou 2018; Hidasi et al. 2015; Jannach and Ludewig 2017; Ludewig and Jannach 2018; Rendle et al. 2009) on the state-of-the-art session-based recommendation.
- We evaluate the efficacy of our approach on three datasets: firstly, a proprietary dataset collected from the online student job portal Studio Jobs; secondly, a publicly available job dataset that was provided by XING after the RecSys Challenge 2017; and thirdly, a publicly available job dataset from the job platform CareerBuilder.
- We train and test the autoencoders on two sources of job-related data: (1) interaction data from sessions and (2) content features of job postings, for which interactions took place during a session. Our results show that variational autoencoders provide competitive job recommendations in terms of accuracy compared to the state-of-the-art session-based recommendation algorithms.
- We additionally evaluate all session-based job recommender approaches in terms of the beyond-accuracy metrics with system-based and session-based novelty as well as coverage. We find that autoencoders can produce more novel and surprising recommendations compared to the baselines and, at the same time, provide relevant jobs for the user while maintaining a high coverage.
- We provide the implementation of our approach as well as a more detailed hyperparameter description in a public GitHub repository⁶ in order to foster reproducible research.

⁶ <https://github.com/lacic/session-knn-ae>.

Organization of the paper The remainder of the paper is structured as follows: In Sect. 2, we discuss related work. Section 3 outlines our approach to employ autoencoders for session-based job recommendation. Section 4 describes the baseline approaches, datasets, evaluation protocol and performance metrics. Section 5 elaborates on the results of our experiments. Finally, in Sect. 6, we conclude the paper and provide an outlook on our plans for future work.

2 Related work

At present, we identify two lines of research that are related to our work: (1) job recommender systems and (2) session-based recommender systems.

Job recommender systems Job recommender systems address a particular recommendation problem, in that a company might want to hire only a few candidates, while classic recommender systems typically recommend items that are relevant for a large number of users (Kenthapadi et al. 2017). There are two directions of the recommendation problem: One is to recommend jobs to a user given her user profile, while the other is to recommend candidates for a job posting. The directions of both problems can even be combined using a reciprocal recommender (Mine et al. 2013).

Research on recommending jobs to users has mostly focused on improving accuracy with methods like collaborative- and content-based filtering or hybrid combinations of both (Al-Otaibi and Ykhlef 2012; Zhang and Cheng 2016). One example of a hybrid job recommendation system that uses interaction data as well as content data is the work of Liu et al. (2017). Here, the recommendation problem corresponds to first searching for matching candidates for a given job and then recommending this job to these candidates. In another job recommender system presented in Hong et al. (2013), the authors propose to first cluster user profiles based on their characteristics and then to design separate recommendation strategies for each cluster.

In 2016, XING (a career-oriented social networking site based in Europe) organized a challenge for the ACM RecSys conference to build a job recommendation system (Abel et al. 2016) that recommends a list of job posts with which a user might interact in the upcoming week. The winning approach (Xiao et al. 2016) used a hierarchical learning-to-rank model to generate the recommendations, which captures semantic relevance, temporal characteristics of a user's profile information, the content of job postings and the complete log of user activities. The anonymized challenge dataset has since been employed, for instance, by Mishra and Reddy (2016), who built a gradient boosting classifier to predict if a given user will like a particular job posting. In 2017, XING organized another recommender challenge for the ACM RecSys conference (Abel et al. 2017). Here, the recommendation problem was turned into a search for suitable candidates when a new job posting is added to the system (i.e., the task constitutes a cold-start problem (Lacic et al. 2015)). The winning approach (Volkovs et al. 2017) spent considerable effort on feature engineering to train a gradient boosting algorithm, which determines the probability of whether or not a given candidate user profile is suited for a target job posting.

In our work, we employ the most recent version of the dataset provided by XING after the RecSys challenge 2017 to evaluate a range of approaches to provide job

recommendations in anonymous sessions. Besides, in our experiments, we use a proprietary dataset gathered from Studio Jobs, an Austrian student job portal, as well as a publicly available dataset from the job portal CareerBuilder.

Since in our work, we focus on session-based job recommendations, in the next paragraph, we summarize related work on session-based recommender systems.

Session-based recommender systems Most recommender systems require a user preference history in the form of explicit or implicit user interactions. Based on the user preference history, a user profile is created, which is the basis for approaches such as matrix factorization (Koren et al. 2009). However, it is not always possible to create such user profiles, e.g., to protect the privacy of users or due to inadequate resources. As a remedy, session-based recommender systems (Hidasi et al. 2015) have been proposed, which model a user's actions within a session, i.e., a short period when the user is actively interacting with the system. A simple approach toward session-based recommendation is to recommend similar items using item–item similarity as proposed by Sarwar et al. (2001). Hidasi and Tikk (2016) propose a general factorization framework that models a session using the average of the component latent item representations. Shani et al. (2005) use Markov decision processes to compute recommendations that incorporate the transition probability between items. Jannach and Ludewig (2017) use co-occurrence patterns as a basis for session-based recommendations. They report comparable and often even a superior performance of a heuristics-based nearest neighbor method (KNN) to generate recommendations in a session-based setting in comparison with competitive, state-of-the-art methods based on neural architectures. Hence, in our work, we also use two KNN-based methods, i.e., sequential session-based KNN and vector multiplication session-based KNN (Ludewig and Jannach 2018) as baseline algorithms due to their good performance and scalability as reported in related works (Jannach and Ludewig 2017; Kamehkhosh et al. 2017; Ludewig and Jannach 2018).

In general, applying neural networks in session-based recommendation systems has gained much attention in recent years. For instance, recent work (Tuan and Phuong 2017; Yuan et al. 2019) uses convolutional networks to produce session-based item recommendations. Song et al. (2016) proposed a neural architecture that combines both long-term and short-term temporal user preferences. They model these preferences through different long short-term memory (LSTM) networks in a stepwise manner. In this vein, Lin et al. (2018) introduce STAMP (short-term attention/memory priority) that simultaneously incorporates a user's general interest (i.e., long-term memory) and current interest (i.e., short-term memory). Wu et al. (2018) present an architecture for session-based recommendations that is based on graph neural networks. Here, using an attention network, each session is also represented by a session user's global preference and their current interest. The authors of Li et al. (2017) propose NARM (neural attentive recommendation machine), which uses an attention mechanism in a hybrid encoder to model the sequential behavior of a user and to extract the user's main purpose from the current session. As the authors show, this approach is specifically well suited to model long sessions.

Out of the different neural architectures, recurrent neural networks have become particularly popular for the task at hand (Chatzis et al. 2017; Hidasi et al. 2015; Smirnova and Vasile 2017). In the earlier mentioned work of Hidasi

et al. (2015), the authors showed that a recurrent neural network (RNN)-based approach can model variable-length session data. Other related papers on sequential data either improve the original algorithm (Hidasi and Karatzoglou 2018; Tan et al. 2016) or extend it by capturing additional information such as context (Twardowski 2016) or attention (Li et al. 2017). In later work, Hidasi et al. (2016) introduce an architecture (i.e., pRNN) that combines multiple RNNs to model sessions via clicks as well as via features of the clicked items such as content information. Here, each RNN handles a particular feature, such as the clicked item's textual representation. The authors show that, given the optimal training strategy, pRNN architectures can result in higher performance compared to feature-less session models. Due to its ability to incorporate content features of job postings in its model in addition to interactions within sessions, in our work, we use pRNN as a baseline approach as we also take into account content features of job postings as well as interactions.

In our work, we employ autoencoders, a type of neural network that can reduce the dimensionality of data (Kramer 1991), to infer latent session representations and to generate recommendations. Specifically, we propose to employ a classic autoencoder (Kramer 1991), a denoising autoencoder (Vincent et al. 2008) and a variational autoencoder (Jordan et al. 1999) to model and encode sessions. In this vein, we find that collaborative denoising autoencoders (CDAE) (Wu et al. 2016) are related to our work. CDAE utilize a denoising autoencoder (Vincent et al. 2008) by adding a latent factor for each user to the input. A denoising autoencoder can learn representations that are robust to small, irrelevant changes in the input. In CDAE, the number of parameters grows linearly with the number of users and items, which makes it prone to overfitting (Liang et al. 2018). Also related to our work is neural collaborative filtering (He et al. 2017), where a neural architecture, which can learn any function from data, replaces the dot product between the latent user and item features. However, this model has a similar issue as CDAE and, thus, grows linearly with the number of sessions and jobs as the authors of Liang et al. (2018) describe.

Finally, with respect to evaluation, to the best of our knowledge, related work on evaluating session-based recommender systems with beyond-accuracy metrics such as system-based and session-based novelty, or coverage is scarce. Only in recent work, Ludewig and Jannach (2018) evaluate session-based recommender systems in light of coverage and popularity bias. With this work, we aim to contribute to this sparse line of research as we evaluate all approaches in this work with respect to system-based and session-based novelty as well as coverage, in addition to accuracy.

3 Approach

In this section, we describe our approach toward a session-based job recommender system using autoencoders. In Sect. 3.1, we first describe how we encode sessions with autoencoders. Then, in Sect. 3.2, we outline our method to model the input session vectors from interactions and content features. Finally, Sect. 3.3 details how we compute session-based job recommendations.

3.1 Encoding sessions using autoencoders

Autoencoders are a type of neural network, which were popularized by Kramer (1991) as a more effective method than principal component analysis (PCA) with respect to describing and reducing the dimensionality of data. Autoencoders are trained in an unsupervised manner where the network is trying to reconstruct the input by passing the information to the output layer through a bottleneck architecture. For our work, we employ three variants of autoencoder architectures to represent a session: (1) a classical autoencoder (AE), (2) a denoising autoencoder (DAE) and (3) a variational autoencoder (VAE).

Autoencoder (AE) The simplest form of an autoencoder has only one hidden layer (i.e., the latent layer) between the input and output (Bengio et al. 2007). The latent layer takes the vector $x_s \in \mathbb{R}^D$, which represents the session and maps it to a latent representation $z_s \in \mathbb{R}^K$ using a mapping function:

$$z_s = h(x_s) = \sigma(W^T x_s + b)$$

where W is a $D \times K$ weight matrix, $b \in \mathbb{K}$ is an offset vector and σ is usually a non-linear activation function. Using z_s , the network provides a reconstructed vector $\hat{x}_s \in \mathbb{R}^D$, which is calculated as:

$$\hat{x}_s = \sigma(W' z_s + b')$$

By adding one or more layers between the input and latent layer, we create an *encoder* and, correspondingly, a *decoder* by doing the same between the latent and output layer, hence the name autoencoder. During inference, we use the output of the latent layer (i.e., the information bottleneck) to represent the latent session vector z_s .

In our experiments, for σ , we use rectified linear units (ReLU⁷) (Nair and Hinton 2010) activation function for all layers except the final output layer, where a sigmoid activation function is used. Furthermore, we use a $D_s - 256 - 100 - 256 - D_s$ network architecture,⁸ where D_s is the dimension of the original vector representation of the session that is encoded using job interactions with or without the corresponding job content data. To train the network, we use RMSprop (Tieleman and Hinton 2012) and minimize the Kullback–Leibler divergence (Fischer and Igel 2012).

We also experimented with adding additional encoder/decoder layers as well as increasing the layer size (e.g., layers with a size of 1000) but did not see any major performance differences besides an increased training complexity. Both Adam and RMSProp are two of the most popular adaptive stochastic algorithms for training deep neural networks. In our work, we focused on RMSProp.

Denoising Autoencoder (DAE) As shown by Vincent et al. (2008), extending autoencoders by corrupting the input can show surprising advantages. The idea of a

⁷ For an input x , $relu(x) = max(0, x)$.

⁸ We also tested higher values for the dimension of the latent layer (e.g., layers with a size of 1000) as well as adding additional encoder/decoder layers, but did not find enough accuracy improvement that would justify the additional computation burden when calculating session similarities in real time.

denoising autoencoder is to learn representations that are robust to small, irrelevant changes in the input. Corrupting the input can be done on either one or multiple layers before we calculate the final output.

In our DAE model, we get a corrupted input \hat{x} using the commonly employed additive Gaussian noise on the input layer with a probability of 0.5. Like earlier, we use the same $D_s - 256 - 100 - 256 - D_s$ architecture, ReLU and sigmoid activation functions, the RMSprop optimization algorithm and the Kullback-Leibler divergence as loss function.

Variational Autoencoder (VAE) Another approach to extract the latent representation z_s is to use variational inference (Jordan et al. 1999). For that, we approximate the intractable posterior distribution $p(z_s|x_s)$ with a simpler variational distribution $q_\phi(z_s|x_s)$, for which we assume an approximate Gaussian form with an approximately diagonal covariance:

$$\log q_\phi(z_s|x_s) = \log \mathcal{N}(z_s; \mu, \sigma^2 I)$$

where μ and σ^2 is the encoded output given the input vector representation x_s of a session. To be more precise, we use additional neural networks as probabilistic encoders and decoders. Most commonly, this is done using a multilayered perceptron (MLP). For the above-mentioned $q_\phi(z_s|x_s)$, we calculate:

$$\begin{aligned}\mu &= W_2 h + b_2 \\ \log \sigma^2 &= W_3 h + b_3 \\ h &= \text{relu}(W_1 x_s + b_1)\end{aligned}$$

where $\{W_1, W_2, W_3, b_1, b_2, b_3\}$ are weights and biases of the MLP and are part of variational parameters ϕ . While decoding, we sample the latent representation and produce a probability distribution $\pi(z_s)$ over all features from the input session vector x_s . As we deal with implicit data, to calculate the probabilities, we let $p_\theta(x_s|z_s)$ be a multivariate Bernoulli (Kingma and Welling 2013), whose probabilities in the MLP we calculate as:

$$\begin{aligned}\log p(x_s|z_s) &= \sum_{i=1}^D x_{si} \log y_i + (1 - x_{si}) \cdot \log(1 - y_{si}) \\ y_s &= f_\theta(W_5 \text{relu}(W_4 z_s + b_4) + b_5)\end{aligned}$$

where f_θ is an element-wise nonlinear activation function (i.e., in our case a sigmoid) and $\theta = \{W_4, W_5, b_4, b_5\}$ are weights and biases of the MLP.

The generative model parameters θ are learned jointly with variational parameters ϕ by optimizing the marginal likelihood of the data. The objective is thus to minimize the distance between the variational lower bound $\mathcal{L}(\theta, \phi, x)$ and a certain prior (Kingma and Welling 2013; Liang et al. 2018), which in case of VAEs is the Kullback–Leibler divergence (Fischer and Igel 2012) of $q_\phi(z_s|x_s)$ and $p(z_s|x_s)$. As we are sampling z_s from q_ϕ in the variational lower bound, in order to learn the model, we need to apply the reparametrization trick (Kingma and Welling 2013; Rezende et al. 2014) by sampling $\epsilon \sim \mathcal{N}(0, I_K)$ (also seen later in Fig. 2) and reparametrize

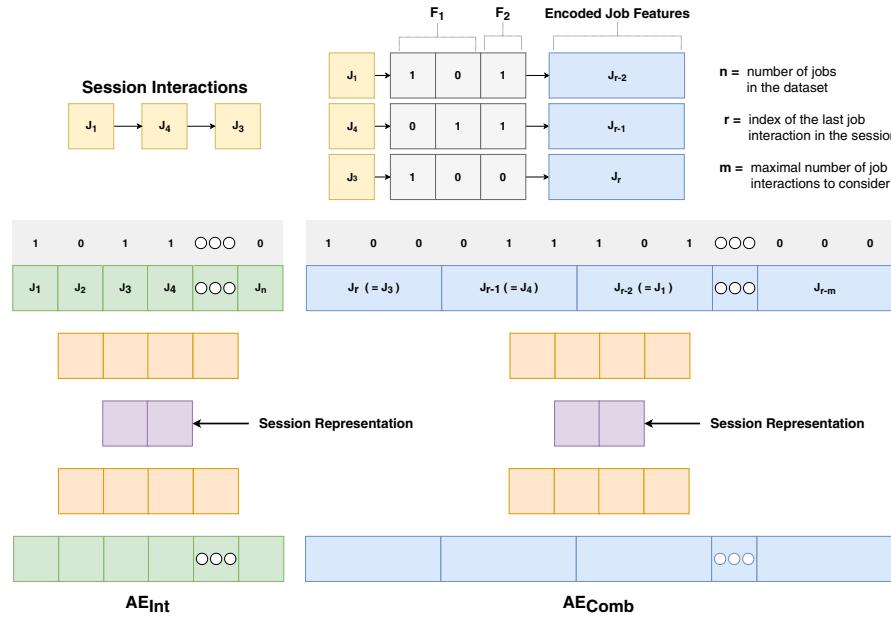


Fig. 1 Modeling session vectors. The input of the utilized autoencoder is a session representation, which can be binary-encoded using job interactions with or without the corresponding job content data. For example, a standard autoencoder that only considers interaction data (denoted as AE_{Int}) will expect a binary encoded vector with a dimension that equals the number of jobs in the underlying dataset. To combine this with job content data (denoted as AE_{Comb}), we use the most recent m job interactions within the session and generate a binary encoding of the job content features in descending order

$z_s = \mu_\phi(x_s) + \epsilon \odot \sigma_\phi(x_s)$. Hence, the gradient with respect to ϕ can be back-propagated through the sampled z_s .

In our experiments, we utilize the described VAE model with a similar architecture as previously mentioned: $D_s - 256 - 100 - 256 - D_s$ (i.e., the encoder and decoder MLPs are symmetrical). Furthermore, for all three autoencoder architectures, we experiment on additionally incorporating the self-attention mechanism (e.g., as Lin et al. 2017; Parikh et al. 2016; Vaswani et al. 2017 do in their work) on the encoder layer.

3.2 Modeling session vectors

The input for any of the three autoencoder variants is a binary-encoded representation of the session x_s . As shown in Fig. 1, we propose the following two variants of how to train the autoencoder models that will be used to infer the latent representation z_s .

Variant 1: Modeling from interactions We construct x_s by only using the job interaction data of a given session. In the remaining paper, we denote the three autoencoder models, which only use job interaction data as AE_{Int} , DAE_{Int} and VAE_{Int} . We create session vectors of size n_s , where n_s is the number of jobs in the

underlying dataset. Each job is then assigned an index in this vector. The interactions on the corresponding job indices are set to 1, while we set the rest to 0. One possible drawback of this approach is that due to the ephemeral nature of job postings, we would need to frequently retrain the utilized model in order to consider new jobs coming to the system (Matuszyk et al. 2015). Moreover, this will also impact the size of the input vector x_s , which will constantly be increasing with every new job.⁹

Variant 2: Modeling from interactions and content In order to mitigate the need to retrain the autoencoder models frequently, we also propose to leverage the content of job postings, with which anonymous users have interacted during a session (i.e., combine interaction data with content data). Given a set of content features $F = \{f_1, \dots, f_l\}$, we first convert each job interaction in a session to a binary vector of size $n_j = \sum_{i=1}^l dist(f_i)$, where $dist(f_i)$ gives the number of distinct values of a job feature f_i . Each feature value is then assigned an index in this vector, and the existing feature values are set to 1, while the rest are 0. To create the session vector x_s , starting from the most recent job interaction, we concatenate the last m converted job interactions. In case the number of job interactions is less than m , x_s is right-padded with 0-filled job vectors, which results in x_s being of size $n_j \times m$. We denote the three autoencoder models that use the content features of job interactions as AE_{Comb} , DAE_{Comb} and VAE_{Comb} . Note also that we introduce the parameter m to end up with an input vector x_s that has a fixed length and a model that is less sensitive to new job postings that are added to the system.

3.3 Computing session-based job recommendations

We formulate the recommendation problem as follows: Given a target session s_t , in which there was an interaction with at least one job j_i from the set of available jobs $J = \{j_1, \dots, j_n\}$, the task is to predict the next jobs this user will likely interact with. In order to compute recommendations, as shown in Fig. 2, we first extract the output z_s for the sessions that are available in the training set. During prediction time, for a given target session s_t , we proceed to infer its latent representation first to find the top- k similar past sessions. In order to reduce the computational burden and allow for efficient recommendation,¹⁰ we extract a subset of all sessions, where the users have interacted with the last job in s_t . Using z_s , we compute the cosine similarity between the respective target and candidate session and use the top- k similar sessions to recommend jobs. Jobs are then ranked based on the following score:

$$sKNN(s_t, j_i) = \sum_{i=1}^n sim(s_t, s_i) \times 1_{s_i}(j_i)$$

⁹ This effect can, however, be damped by removing obsolete job postings, but would still result in a constantly changing input dimension.

¹⁰ The number of stored sessions can easily pass the million mark and cause for unnecessary calculations once a recommender system is running for a longer period.

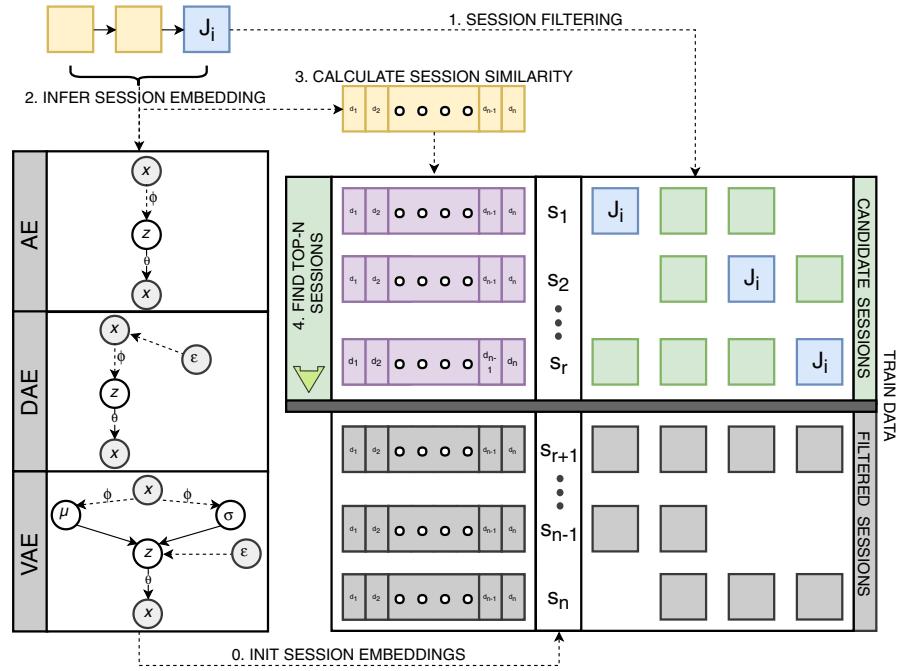


Fig. 2 Computing session-based job recommendations. Using the trained autoencoders, we infer latent representation for (1) sessions in the training data and (2) the current target session for which we recommend jobs. Jobs from the top-k similar candidate sessions (filtered by the currently interacted job posting) are recommended to the target session

where $1_{s_i}(j_i)$ is 1 if the candidate session s_i contains the job j_i and 0 otherwise (as in Bonnin and Jannach 2015; Jannach and Ludewig 2017).

4 Experimental setup

In this section, we present the baseline approaches and the datasets we used for this study. We outline the evaluation protocol and the performance measures, which we employed to compare all approaches. In our evaluation, we contribute to the limited amount of related work (e.g., like Ludewig and Jannach 2018) as we evaluate all approaches both concerning accuracy and beyond-accuracy measures (i.e., system-based and session-based novelty as well as coverage).

4.1 Baseline approaches

We utilize well-known baselines and compare our approach to the following state-of-the-art methods (Ludewig and Jannach 2018) for session-based recommendation:

POP A simple and yet often strong baseline for session-based recommendation is the popularity-based approach. As in Hidasi et al. (2015), the results are always the same top- k popular items from the training dataset.

iKNN The item-KNN approach recommends jobs that are similar to the actual job that is interacted with during the session. As in Hidasi et al. (2015), we use the cosine similarity and include regularization to avoid coincidental high similarities between rarely visited jobs.

BPR-MF One of the commonly used matrix factorization methods for implicit feedback is Bayesian personalized ranking (Rendle et al. 2009). As in Hidasi et al. (2015), we use the average of job feature vectors of the jobs that had occurred in the current session as the user feature vector to apply it directly to generate a session-based recommendation. That is, similarities of the feature vectors are averaged between a candidate job and the jobs of the current session.

Bayes Following the Bayesian rule, we calculate the conditional probability of a job x_i being clicked based on the previous r interactions of the current session s :

$$P(x_i|x_{s_1}, \dots, x_{s_r}) = \frac{\prod_{j=1}^r P(x_{s_j}|x_i) \times P(x_i)}{\prod_{j=1}^r P(x_{s_j})}$$

This approach is, from a computational perspective, inexpensive to calculate and run in an online setting.

GRU4Rec Recently, Hidasi et al. (2015) showed that recurrent neural networks are excellent models for data generated in anonymous sessions. GRU4Rec combines gated recurrent units with a session-parallel mini-batch training process, and it incorporates a ranking-based loss function. For our study, we use the most recent improvement in GRU4Rec (Hidasi and Karatzoglou 2018). This GRU4Rec version employs a new class of loss functions tied together with an improved sampling strategy.

pRNN Another recent advancement of Hidasi et al. (2016) shows how to incorporate item features into the representation of neural networks. They propose several different architectures based on GRU units and ways to train them. We use a parallel architecture with simultaneous training for our experiments. This approach utilizes both a one-hot encoding of the current item interaction and an item representation as inputs for the subnets. The trained model uses the TOP1 loss function as defined in Hidasi et al. (2015).

sKNN Recent research has shown that computationally simple nearest-neighbor methods can be effective for session-based recommendation Jannach and Ludewig (2017). The session-based KNN approach first determines the k most similar past sessions in the training data. Sessions are encoded as binary vectors of the item space, and a set of k nearest sessions is retrieved for the current session using cosine similarity. The final job score is calculated by aggregating the session similarity over all the sessions that contain the candidate job.

V-sKNN Vector multiplication session-based KNN (V-sKNN) is a variant of sKNN that considers the order of the elements in a session. The idea here is to create a real-valued vector by putting more weight on recent interactions, where

Table 1 Statistics of the datasets Studio, RecSys Challenge 2017 (i.e., RecSys17) and CareerBuilder12

Dataset	# Interactions	# Sessions	# Jobs	Sparsity (%)
Studio	191,259	26,785	1111	99.36
RecSys17	55,380	16,322	15,686	99.98
CareerBuilder12	661,910	120,147	197,590	99.99

While Studio has more sessions and job interactions, the RecSys Challenge 2017 dataset has more job postings that can be recommended. CareerBuilder12 is the largest dataset, but also has the highest sparsity

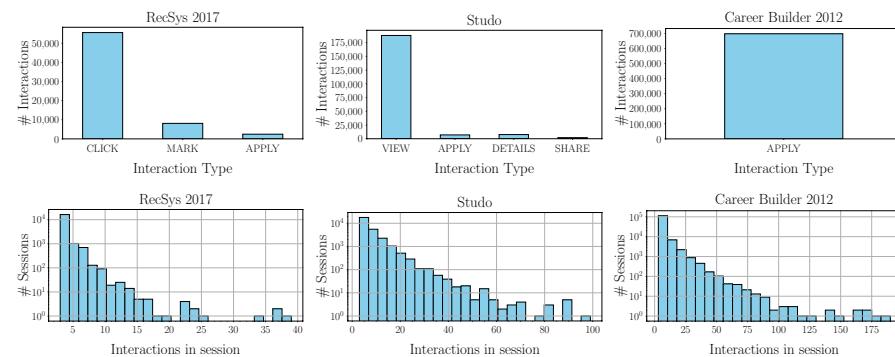


Fig. 3 Number of interactions based on the interaction type (top) and the distribution of session sizes (bottom) is shown for the RecSys Challenge 2017 (left) and Studio (middle) and CareerBuilder12 datasets. Overall, the distribution of interaction types is similar between the datasets where the *click*, *view* and *apply* interactions mostly dominate

only the very last element of the session obtains a value of “1” (Ludewig and Jannach 2018). For this, a linear decay function is used that depends on the position of an element within the session.

S-sKNN Sequential session-based KNN (S-sKNN) puts more weight on elements that appear later in the session in a similar way as V-sKNN (Ludewig and Jannach 2018). This effect is, however, achieved by giving more weight to neighboring sessions which contain recent items of the current session.

4.2 Datasets

For this study, we employ three different datasets from the job domain. The first dataset, *Studio*, is a proprietary dataset collected from the online platform Studio Jobs, a job-seeking service for university students. The second dataset *RecSys17* is the latest version of the data provided by XING after the RecSys Challenge

Table 2 Binary-encoded content features of our three datasets

Studio		RecSys17		CareerBuilder12	
Content feature	Encoding	Content feature	Encoding	Content feature	Encoding
Job state [†]	10	Region [†]	17	State [†]	55
Job country [‡]	1	Country [‡]	4	Requirement topic	20
Job begins now	1	Is payed	2	Title topic	20
Job effort	1	Career level	6	Description topic	20
Job language	1	Industry Id ^{††}	23		
Job discipline ^{††}	40	Discipline Id ^{††}	22		
Employment type ^{‡‡}	6	Employment ^{‡‡}	5		

For Studio, concatenating all job features results in a job vector with a dimensionality of 60. For the RecSys17 dataset, this results in a job vector with a dimension of 79. For the CareerBuilder12 dataset, this results in 115 dimensionality vectors. We also put the same annotation on content features, which have a similar meaning in both datasets. The *Job Discipline* feature is the only one in Studio, which represents a combination of the *Discipline Id* and *Industry Id* features from the RecSys17 dataset

2017 (Abel et al. 2017). The third dataset *CareerBuilder12* is from an open Kaggle competition, called Job Recommendation Challenge,¹¹ provided by the online employment Web site CareerBuilder. The statistics of all three datasets are given in Table 1. As seen, all datasets have a high sparsity: 99.36% for Studio, 99.98% for RecSys17 and 99.99% for CareerBuilder12. Studio contains a higher number of sessions when compared to RecSys17 but has a much smaller number of available jobs that can we can recommend. CareerBuilder12 is the largest dataset of the three, but only contains job applications as interactions. In the next paragraphs, we describe the three datasets in more detail.

Studio The dataset contains job interactions from anonymous user sessions from a period of three months between September 2018 and December 2018. All job interactions in this dataset have an anonymous session id assigned to them. As seen in the top row of Fig. 3, the Studio dataset contains four interaction types, i.e., job *view*, show company *details*, *apply* and *share job*. As shown at the bottom row of Fig. 3, the log histogram of session sizes follows a skewed pattern, which means that most sessions have a small number of interactions. In particular, every session has 6.98 interactions on average and a median of 5 interactions.

Concerning content features reported in Table 2, in the Studio dataset, we utilize seven content features of job postings. The *Job State* determines 1 out of 9 Austrian federal states. *Job Country* indicates whether the job is in Austria or some other country. The *Job Begins Now* feature specifies whether the job candidate can start immediately working on the advertised position. We relate this feature to the *Is Payed* feature from the RecSys17 dataset as companies typically pay for job postings to be shown if they urgently need candidates. Studio's *Job State* is similar to the *Region* feature from RecSys17, the same holds true for Studio's *Employment Type*,

¹¹ <https://www.kaggle.com/c/job-recommendation>.

which can be related to the *Employment* feature from RecSys17. *Job Effort* indicates whether the concrete working hours are specified; otherwise, the default working hours are assumed. The *Job Language* feature specifies whether the job requires the usage of either the German or English language. Furthermore, a job posting can also be described by a subset of 40 different *Job Discipline* labels and a subset of 6 different *Employment Type* labels. The *Job Discipline* feature can actually be regarded as a combination of the *Discipline Id* and *Industry Id* features from the RecSys17 dataset. As described in Sect. 3.2, we use all content features from the Studio dataset to create a binary-encoded job vector with a dimensionality of 60. Finally, we have 77.7% uniquely encoded job vectors, which consist, on average, of 11.8% assigned feature values.

RecSys17 The dataset contains six different interaction types that were performed on the job items. For this study, we only keep the *click*, *bookmark* and *apply* interactions (as seen on the top of Fig. 3). We remove the *delete recommendation* and *recruiter interest* interactions as these are irrelevant in our setting. Moreover, we discard *impression* interactions as they are created when XING shows a job to a user. As stated by Bianchi et al. (2017), an impression does not imply that the user has interacted with the job. The dataset consists of interactions from a period of three months (from November 6, 2016, until February 3, 2017). We manually partition the interaction data of the RecSys dataset into sessions using a 30-minute idle threshold (as in Quadrana et al. 2017). The resulting sessions have, on average, 3.62 interactions per session and a median of 3 interactions.

Also, the RecSys17 dataset contains content features about the job postings, such as career level or type of employment. From this set, we select seven features as content-based input for our approaches and discarded the numeric IDs of title and tags, since those would lead to very big encodings. The chosen features closely resemble the features that are present in the Studio dataset. More specifically, from RecSys17, we use the following features, as shown in Table 2: *Region*, *Employment*, *Is Payed*, *Discipline Id*, *Career Level*, *Industry Id* and *Country*. The *Region* content feature is a categorical feature with 17 possible values, like the *Employment* feature with 5 values. The *Is Payed* content feature indicates if the posting is a paid for by a company. The *Discipline Id* is a categorical feature with 22 different values that represent disciplines such as consulting or human resources. The categorical feature *Career Level* can take 7 values, *Industry Id* represents industries such as finance, and *Country* denotes the code of the country in which the job is offered.¹² Overall, we end up with a job vector that has dimensionality of 79. We find that only 33.58% of the job vectors are unique and have on average 8.86% assigned feature values.

CareerBuilder12 The dataset contains job applications from a period of almost three months. No other interaction types are present in this dataset. The sessions are created via a time-based split of 30 min. Due to the nature of job applications, most sessions contain very few interactions. The interactions in the dataset happen over 13 weeks. Thus, similar to the other two datasets, it happens over almost three

¹² <https://www.recsyschallenge.com/2017>.

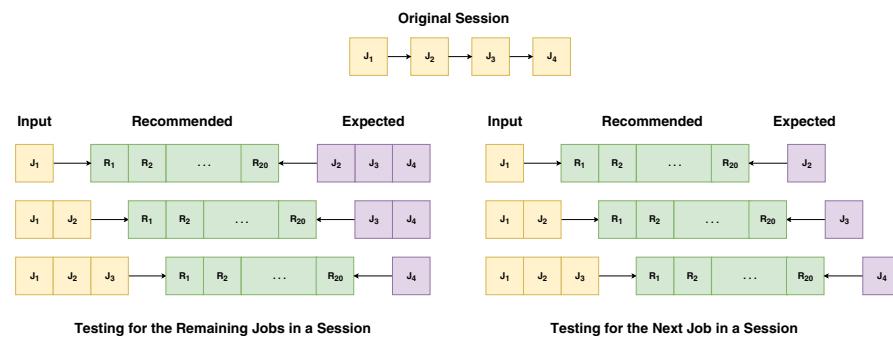


Fig. 4 Our evaluation protocol for one exemplary session consisting of four jobs. We distinguish between (1) comparing the recommended jobs with the remainder of the interactions in a session (left) and (2) comparing the recommended jobs with the next job interaction (right)

months, i.e., from April 2012 to June 2012. The sessions have, on average, 5.64 job applications per session, whereas the median is 4 applications per session.

Regarding content features, the CareerBuilder12 dataset contains textual descriptions of the jobs as well as categorical data for the location. From the content data, the 55 different states are used in the form of one-hot encodings. Since in our work, we utilize categorical job features as input to our models, we additionally inferred categorical topics for each of the 3 textual features (i.e., title, description and requirements). That is, for every textual feature, we trained a separate latent Dirichlet allocation (LDA) model from which we extracted 20 distinct topics. This procedure resulted in every job posting having a requirement, title and description topic assigned to them. Thus, the resulting feature vector of a job posting is of size 115. For this largest dataset, 13.46% of vectors are unique, and those vectors have only 2.58% assigned feature values.

4.3 Evaluation protocol

We employ a time-based split on all three datasets to create train and test sets. For this, we put the sessions from the last 14 days (i.e., 2 weeks) in the test set of the respective dataset and use the remaining sessions for training. For each set, we keep only sessions with a minimal number of 3 interactions.¹³ Like (Quadrana et al. 2017), we filter items in the test set that do not belong to the train set as this enables a better comparison with model-based approaches (e.g., RNNs), which can only recommend items that have been used to train the model. In Studio, this procedure results in 23,738 sessions to train and 3047 to test the approaches. For the RecSys17 dataset, this results in 12,712 sessions for training and 3610 sessions for testing. In

¹³ We chose 3 for the minimum amount of interaction as it is the lowest median of interactions in a session across all three datasets, as reported in Sect. 4.2.

the case of the much larger CarrerBuilder12 dataset, the train set contains 108, 783 sessions, whereas the test set has 11, 364 sessions.

Training and testing the algorithms We first train all approaches on the respective training data. In order to evaluate the performance of the utilized session-based recommendation algorithms, for each session in the test data, we iteratively subsample its interactions. That is, after each session interaction, we recommend 20 jobs for the current target session state and compare the predictions with the remaining interactions. We start this procedure for every session after the first interaction and end before the last one. In this setting, as shown in Fig. 4, we explore two evaluation cases: comparing the recommended jobs with (1) the remainder of the interactions in the session and (2) with the next job interaction (i.e., next item prediction; same as in Hidasi and Karatzoglou 2018; Hidasi et al. 2015).

For our proposed method, that uses content features in combination with user interactions to encode the input for the autoencoders (i.e., as described in Sect. 3.2), we use the top 25 recent job interactions to infer the session representation. That is, we set the parameter $m = 25$ as more than 98% of all sessions in Studio, and almost all sessions in the RecSys17 dataset do not have more than 25 job interactions (i.e., as shown in Fig. 3).

Hyperparameter optimization To optimize hyperparameters, we further split the train sets by the same time-based split to generate validation sets. Thus, we use the last 2 weeks of the train set as a separate validation set and the remaining sessions to train our models. The resulting split for the Studio dataset is 19, 245 sessions in the validation train set and 3273 sessions in the validation test set. For the RecSys17 dataset, we have 8001 sessions in the validation train set and 2046 sessions in the validation test set. In case of CareerBuilder12, the validation train set contains 51, 717 sessions and the validation test set 10, 574 sessions. Note that some sessions did no longer have the minimal number of 3 interactions and were filtered out. As a consequence, the combination of the validation train and validation test set is smaller than the original train set. The results of the hyperparameter optimization step are described in Sect. 5.3.

4.4 Evaluation metrics

We quantify the recommendation performance of each approach concerning accuracy and beyond-accuracy metrics like system-based and session-based novelty. More specifically, in our study, we use the following performance measures:

Normalized Discounted Cumulative Gain (nDCG) nDCG is a ranking-dependent metric that measures how many jobs are predicted correctly. Also, it takes the position of the jobs in the recommended list into account (Parra and Sahebi 2013). It is calculated by dividing the DCG of the session's recommendations with the ideal DCG value, which is the highest possible DCG value that can be achieved if all the relevant jobs would be recommended in the correct order. The nDCG metric is based on the *Discounted Cumulative Gain (DCG@k)*, which is given by Parra and Sahebi (2013):

$$DCG@k = \sum_{i=1}^k \frac{2^{rel(i)} - 1}{log(1 + i)}$$

where $rel(i)$ is a function that returns 1 if the recommended job at position i in the recommended list is relevant. $nDCG@k$ is calculated as $DCG@k$ divided by the ideal DCG value $iDCG@k$, which is the highest possible DCG value that can be achieved if all the relevant jobs would be recommended in the correct order. Over all the sessions, it is given by:

$$nDCG@k = \frac{1}{|S|} \sum_{s \in S} \left(\frac{DCG@k}{iDCG@k} \right)$$

Mean reciprocal rank (MRR) MRR is another metric for measuring the accuracy of recommendations and is given as the average of the reciprocal ranks of the first relevant job in the list of recommended jobs, i.e., 1 for the first position, $\frac{1}{2}$ for the second position, $\frac{1}{3}$ for the third position and so on. This means that a high MRR is achieved if relevant jobs occur at the beginning of the recommended jobs list (Voorhees 1999). Formally, it is given by Aggarwal (2016):

$$MRR@k = \frac{1}{|S|} \sum_{s \in S} \frac{1}{|H_s|} \sum_{H_j \in H_s} \frac{1}{rank(H_j, R_k)} \quad (1)$$

Here, H_s is the history of the current session s and $rank(H_j, R_k)$ is the position of the first relevant job H_j in the recommended job list R_k .

System-based novelty (EPC) System-based novelty denotes the ability of a recommender to introduce sessions to job postings that have not been (frequently) experienced before in the system. A recommendation that is accurate but not novel will include items that the session user enjoys, but (probably) already knows. Optimizing system-based novelty has been shown to have a positive, trust-building impact on user satisfaction (Pu et al. 2011). Moreover, system-based novelty is also an important metric for the job domain since applying to popular jobs may decrease a user's satisfaction due to high competition and less chance of getting hired (see, e.g., Kenthapadi et al. 2017). In our experiments, we measure the system-based novelty using the expected popularity complement (EPC) metric introduced by Vargas and Castells (2011). In contrast to solely popularity-based metrics (e.g., Zhou et al. 2010), EPC also accounts for the recommendation rank and the relevance for the current session. Thus, the system-based novelty $nov_{system}(R_k|s)$ for the recommendation list R_k of length k for session s is given by:

$$EPC@k = \frac{1}{|S|} \sum_{s \in S} \frac{1}{|R_k|} \sum_{R_i \in R_k} disc(i)p(rel|R_i, s)(1 - p(seen|R_i))$$

Here, $disc(i)$ is a discount factor to weight the recommendation rank i [i.e., $disc(i) = 1/\log_2(i + 1)$] and $p(rel|R_i, s)$ is 1 if the recommended job R_i is relevant for session s or 0 otherwise (i.e., only jobs that are in the current session history are taken

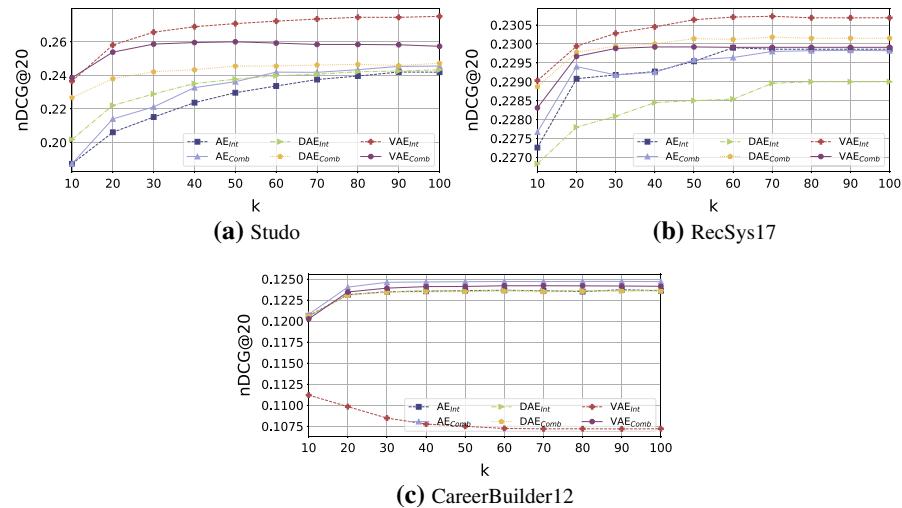


Fig. 5 The figures show the influence of the neighborhood size k for picking top- k similar sessions when comparing the three autoencoder variations on both interaction data and combined data. We find that the recommendation accuracy converges when k is picked to be around 60 or more

into account). Finally, $p(seen|R_i)$ defines the probability that a recommended job R_i was already seen in the system, i.e., $p(seen|R_i) = \log_2(\text{pop}_{R_i} + 1)/\log_2(\text{pop}_{MAX} + 1)$.

Session-based novelty (EPD) In contrast to system-based novelty, session-based novelty incorporates the semantic content of jobs and represents how *surprising* or *unexpected* the recommendations are for a specific session history (Zhang et al. 2012). Given a distance function $d(H_i, H_j)$ that represents the dissimilarity between two jobs H_i and H_j , the session-based novelty is given as the average dissimilarity of all job pairs in the list of recommended jobs R_k and jobs in the current session history H_s (Zhou et al. 2010). In our experiments, we use the cosine similarity to measure the dissimilarity of two job postings using a raw job vector, which contains 1 if a session interacted with it and 0 otherwise. Again, we use the definition by Vargas and Castells (2011) that takes the recommendation rank as well as the relevance for the current session into account. Hence, we measure the session-based novelty $\text{nov}_{session}(R_k|s)$ for the recommendation list R_k of length k for session s by the expected profile distance (EPD) metric:

$$\text{EPD}@k = \frac{1}{|S|} \sum_{s \in S} \frac{1}{|R_k||H_s|} \sum_{R_i \in R_k} \sum_{H_j \in H_s} \text{disc}(i)p(\text{rel}|R_i, s)d(R_i, H_j)$$

Here, H_s is the current history of a session s and $\text{disc}(i)$ as well as $p(\text{rel}|R_i, s)$ are defined as for the EPC metric for measuring the system-based novelty.

Coverage With coverage (Adomavicius and Kwon 2012; Ludewig and Jannach 2018), we assess how many jobs a recommender approach can cover with its predictions. As such, we additionally report the job coverage of each evaluated algorithm. We define the coverage as the ratio between the jobs that have been recommended

and jobs that would be available for recommendation. Here, we make a distinction between coverage types and report the job coverage (1) on the full dataset, i.e., how many of all available jobs can we recommend, and (2) on the test dataset, i.e., how many of the jobs can we recommend that we expect to be interacted with during a session.

5 Results

In this section, we present our experimental results. We first compare the performance of the respective models when used in a k-nearest neighbor manner and then analyze the embedding space of the best-performing autoencoder model. After that, we show the best hyperparameter configurations used for the baseline approaches and then discuss the performance of our approach compared to these baselines.

5.1 Comparing the recommendation performances of autoencoders

We compare the recommendation performance of all three variants of autoencoders, i.e., AE, DAE, VAE, trained on interactions as well as on content. This results in six autoencoder variants in total. We train all autoencoder models for a maximum of 50 epochs or until the error on the validation test set converges. We made additional experiments and incorporated the self-attention mechanism on the encoder layer (e.g., as in Lin et al. 2017; Parikh et al. 2016; Vaswani et al. 2017). We did not find any major improvements, so we do not report the results of these 6 additional autoencoder models.

Figure 5 shows the results of the autoencoder comparison in terms of $nDCG@20$. We compare the results across different values for the neighborhood size k , ranging from 10 to 100. We find that VAE_{Int} , which only uses interactions to encode the input vector, outperforms all other approaches on the Studio and RecSys17 datasets. When combining interaction data with content features (i.e., AE_{Comb} , DAE_{Comb} and VAE_{Comb}), VAE_{Comb} performs the best on the Studio dataset and slightly worse than DAE_{Comb} on the RecSys17 dataset. For the CareerBuilder12 dataset, all approaches except the VAE_{Int} approach have a similar performance. Such accuracy performance for VAE_{Int} suggests that having a much larger item space can be problematic for the generative autoencoder variant. As the variational autoencoder outperforms the other approaches in the majority of the configurations, in the next step, we compare it to the baseline methods. Furthermore, we find that for all autoencoders, accuracy converges after $k = 60$. Thus, in Sect. 5.4, we report the results of VAE_{Int} and VAE_{Comb} using top-60 similar sessions for recommendation.

5.2 Embedding analysis

To better understand the autoencoder models' actual effectiveness, we employ the t-SNE algorithm (Maaten and Hinton 2008) to visualize the embedding spaces. The t-SNE method enables us to visualize high-dimensional data. It reduces the

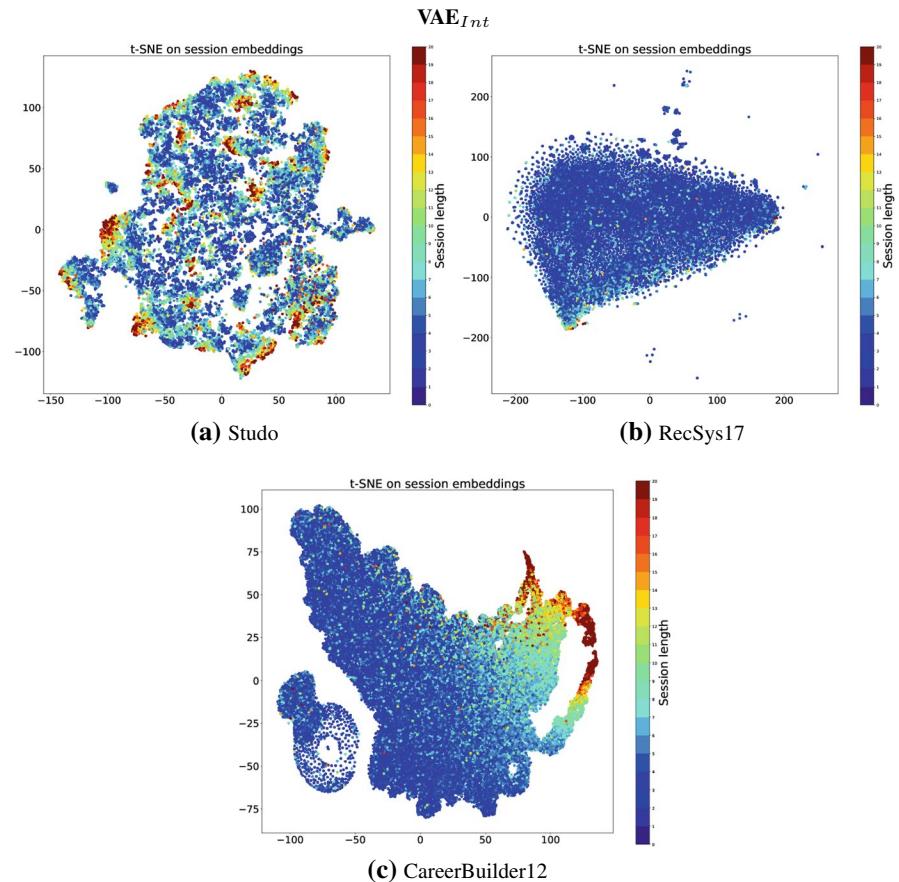


Fig. 6 The plots show t-SNE embeddings for latent session representations produced with the VAE autoencoder models trained on all three datasets using only interaction data. The colors of the sessions reflect the session length, where the same red color is used for sessions with 20 or more interactions

dimensionality of the latent session representations and lets us explore embeddings in a 2D space. In t-SNE plots, similar items are modeled by neighboring points with high probability. In our case, we expect similar sessions to form clusters of neighboring points in the 2D space.

Figures 6 and 7 show the variational autoencoder models as t-SNE plots for all three datasets, i.e., VAE_{Int} trained on interactions and VAE_{Comb} trained on interactions combined with job content (see “Appendix B” for a more detailed embedding analysis). In the case of the smallest dataset Studo, when we train the autoencoder only on interactions, more clusters are produced with sessions of different sizes close to each other (e.g., Fig. 6a). If the variational autoencoders are additionally trained on the job content, we can observe rainbow-colored shapes that are based on session length (e.g., as shown in Fig. 7a, b). In the larger CareerBuilder12 dataset, we end up with several sub-clusters that exhibit

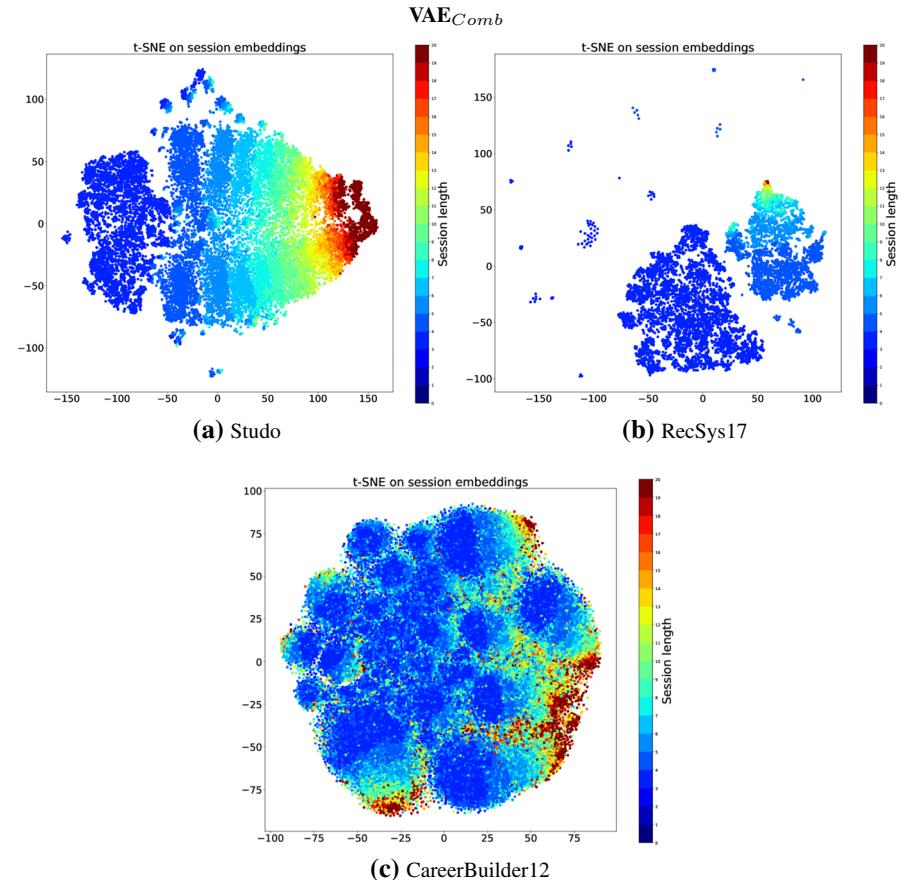


Fig. 7 The plots show t-SNE embeddings for latent session representations produced with the VAE autoencoder models trained on all three datasets using interaction data combined with the job content. The colors of the sessions reflect the session length, where the same red color is used for sessions with 20 or more interactions

this rainbow pattern. In other words, when we encode the input with content features, sessions of similar length tend to cluster. We attribute this to sessions of similar length having similar patterns of input vectors (e.g., many right-padded zeros for short sessions).

Next, we investigate the difference in recommendation accuracy between VAE_{Int} and VAE_{Comb} in light of the clustering patterns. The results suggest that when sessions cluster by similar size in the 2D space, as in the case of VAE_{Comb} in the Studo and RecSys17 datasets and VAE_{Int} in the CareerBuilder12 dataset, recommendation accuracy drops.

Table 3 Best performing hyperparameter settings for each evaluated baseline approach and dataset based on nDCG@20

Approach	Parameter	Studo	RecSys17	CareerBuilder12
BPR	$\lambda_{SESSION}$	0.25	0	0
	λ_{ITEM}	0.25	0	0
iKNN	λ	80	50	20
	α	0.75	0.75	0.75
sKNN	k	100	500	1000
	SAMPLING	Recent	Random	Random
	SIMILARITY	Cosine	Cosine	Jaccard
	POPULARITY BOOST	No	No	Yes
S-sKNN	k	100	500	1000
	SAMPLING	Recent	Random	Random
	SIMILARITY	Cosine	Jaccard	Cosine
	POPULARITY BOOST	No	No	Yes
V-sKNN	k	100	100	100
	SAMPLING	Recent	Random	Random
	SIMILARITY	Cosine	Cosine	Cosine
	POPULARITY BOOST	No	No	No
GRU4Rec	WEIGHTING	Quadratic	Quadratic	Logarithmic
	LOSS	top1-max	bpr-max-0.5	top1-max
	LAYERS	[100]	[100]	[1000]
	DROPOUT	0.2	0.2	0.2
pRNN	BATCH SIZE	32	32	32
	ACTIVATION	tanh	tanh	softmax
	LAYERS	[1000]	[100]	[1000]
	α	0.001	0.01	0.001
	BATCH SIZE	512	512	512

5.3 Hyperparameter optimization of the baseline approaches

We conducted a grid search on the hyperparameters for the baseline approaches using the validation set, i.e., two weeks of user interactions. As such, Table 3 reports on the best performing configurations for each approach and dataset in terms of recommendation accuracy (see “Appendix A” for more details).

BPR We performed a grid search that includes three different values for the regularization of session features $\lambda_{SESSION} \in \{0.0, 0.25, 0.5\}$ and the regularization of item feature $\lambda_{ITEM} \in \{0.0, 0.25, 0.5\}$.

iKNN For the iKNN approach, we evaluated the values for regularization (i.e., to avoid coincidental high similarities of rarely visited items) $\lambda \in \{20, 50, 80\}$ and the normalization factor for the support between two items $\alpha \in \{0.25, 0.5, 0.75\}$.

sKNN, S-sKNN and V-sKNN For all the sKNN variations that we utilize in this paper, we conducted a grid search for the parameter k (i.e., 100, 200, 500 or 1000),

Table 4 Prediction results ($k = 20$) of remaining jobs that will be subject to interaction within a session. (Color table online)

Studio						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	.2724	.1636	.0214	.0174	70.12	100
VAE _{Comb}	.2593	.1597	.0198	.0162	82.27	100
sKNN	.2552	.1403	.0193	.0165	55.18	99.67
V-sKNN	.2766	.1679	.0209	.0178	60.67	100
S-sKNN	.2687	.1532	.0203	.0172	56.26	99.67
GRU4Rec	.2909	.1682	.0230	.0202	53.74	99.02
pRNN	.0895	.0500	.0060	.0058	20.70	29.97
Bayes	.1560	.0758	.0138	.0144	72.73	100
iKNN	.1718	.0864	.0153	.0158	62.65	99.67
BPR-MF	.0700	.0485	.0059	.0028	40.59	69.71
POP	.0501	.0276	.0014	.0048	1.80	2.61
RecSys Challenge 2017						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	.2307	.1962	.0468	.0070	35.82	90.68
VAE _{Comb}	.2299	.1948	.0466	.0068	35.84	90.53
sKNN	.2180	.1782	.0129	.0033	36.33	91.03
V-sKNN	.1931	.1521	.0115	.0033	36.24	88.93
S-sKNN	.2221	.1829	.0131	.0034	36.91	91.43
GRU4Rec	.1040	.0811	.0065	.0024	46.63	73.45
pRNN	.1024	.0593	.0003	.0021	7.87	11.17
Bayes	.0446	.0336	.0031	.0019	28.56	61.62
iKNN	.0565	.0414	.0038	.0024	35.04	70.59
BPR-MF	.2530	.1900	.0086	.0028	76.58	93.79
POP	.2294	.2278	.0001	.0047	0.13	0.30
CareerBuilder 2012						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	.1072	.0393	.0194	.0160	18.20	96.98
VAE _{Comb}	.1242	.0438	.0209	.0177	16.04	96.62
sKNN	.1406	.0454	.0161	.0146	13.77	92.45
V-sKNN	.1458	.0566	.0173	.0154	16.06	95.67
S-sKNN	.1428	.0462	.0166	.0150	14.61	95.03
GRU4Rec	.1415	.0554	.0172	.0159	20.46	83.60
pRNN	.0005	.0002	.0001	.0001	0.02	0.14
Bayes	.0842	.0383	.0094	.0077	11.87	78.53
iKNN	.1386	.0577	.0164	.0144	16.61	90.13
BPR-MF	.0005	.0001	.0001	.0001	78.75	95.36
POP	.0004	.0001	.0001	.0001	0.01	0.08

Table 4 (continued)

Coverage is reported for the ratio of recommended jobs compared to all jobs available in the data set (left) and jobs expected in the test set (right)

the sampling method of sessions (i.e., recent or random), the similarity function (i.e., cosine or Jaccard) and if popular items from neighboring sessions should be boosted. For V-sKNN, we also optimized the decay weighting function (i.e., division, logarithmic or quadratic).

GRU4Rec In the case of GRU4Rec, we experimented with two different loss functions $\{top1\text{-}max, bpr\text{-}max\text{-}0.5\}$, four variations of the number of GRU layers and their sizes $\{[100], [100, 100], [1000], [1000, 1000]\}$, a dropout applied to the hidden layer of $\{0.0, 0.2, 0.5\}$ and batch sizes of $\{32, 128, 512\}$.

pRNN For the pRNN approach, we explored two activation functions $\{\text{softmax}, \tanh\}$ for the output layer, two sizes for the GRU layers $\{[100], [1000]\}$, a learning rate $\alpha \in \{0.01, 0.001\}$ and batch sizes of $\{32, 128, 512\}$. With respect to the batch size, however, due to the computational complexity of pRNN and the size of CareerBuilder12, we were only able to tune this hyperparameter for Studio and RecSys17. As we received the best results for a batch size of 512 for both datasets, we also used a batch size of 512 in case of CareerBuilder12.

5.4 Comparison with baseline approaches

Table 4 shows the results of comparing VAE_{Int} and VAE_{Comb} with all baseline methods when we evaluate against the remaining jobs in the session. We report recommendation accuracy in terms of nDCG and MRR, as well as a system-based novelty (EPC), session-based novelty (EPD) and coverage. In the case of the next job prediction problem, in Figs. 8 and 9, we show nDCG and EPC results for different values of k (i.e., number of recommended jobs).

Accuracy (nDCG & MRR) On all datasets, the $sKNN$ -based approaches achieve high accuracy in terms of nDCG and MRR, as shown in Table 4. In terms of both nDCG and MRR, VAE_{Int} performs second best in RecSys17, while it performs third best in Studio. For the Studio dataset, *GRU4Rec* has the highest accuracy for both metrics. In the RecSys17 dataset, *BPR-MF* performs best concerning nDCG, while *POP* performs best in terms of MRR. In CareerBuilder12, *V-sKNN* achieves the highest nDCG, while *iKNN* achieves the highest MRR. In this dataset, VAE_{Int} achieves medium performance, which we attribute to the ample item space and sparsity of CareerBuilder12. The VAE_{Comb} method, however, results in a higher recommendation accuracy, while training the model is much less expensive.

While the performance of $sKNN$ -based approaches is rather stable, several baselines algorithms, namely *POP*, *BPR-MF*, *iKNN*, *Bayes*, *GRU4Rec* and *pRNN*, show

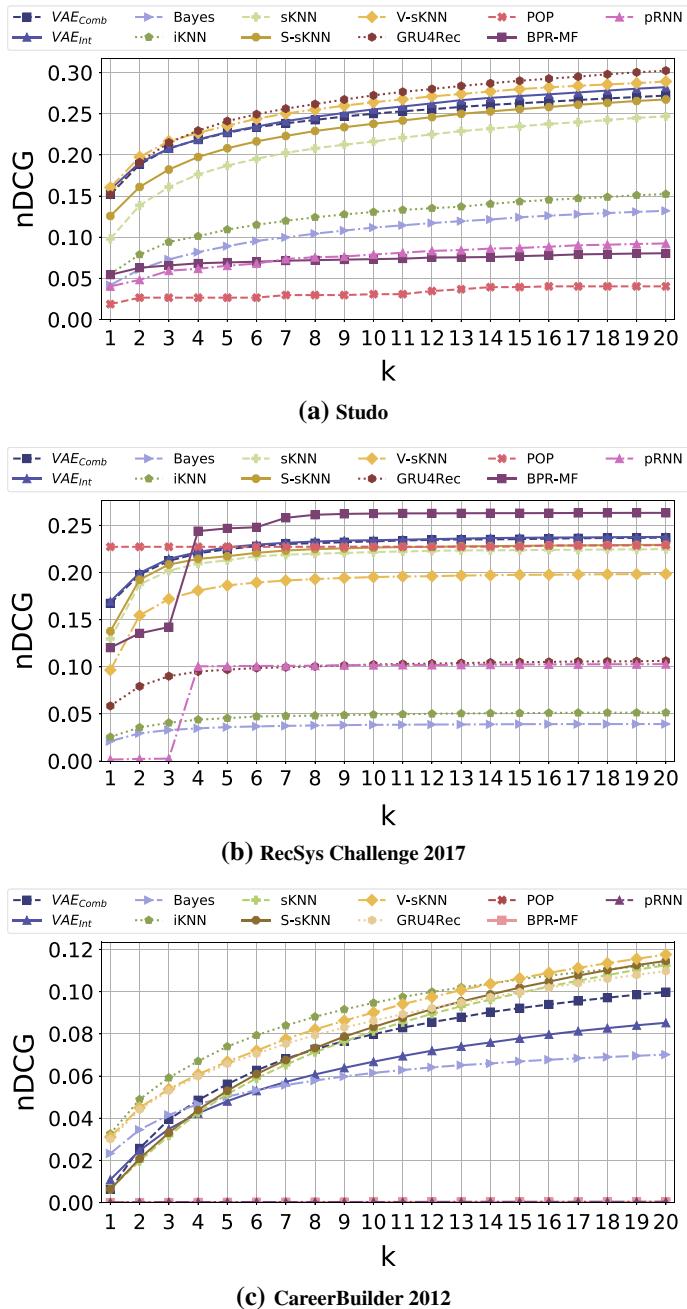


Fig. 8 nDCG results for different recommendation list sizes (i.e., values of k) when predicting the next job in the session. On all three datasets, both our proposed VAE approaches achieve competitive results concerning accuracy (i.e., nDCG) metrics

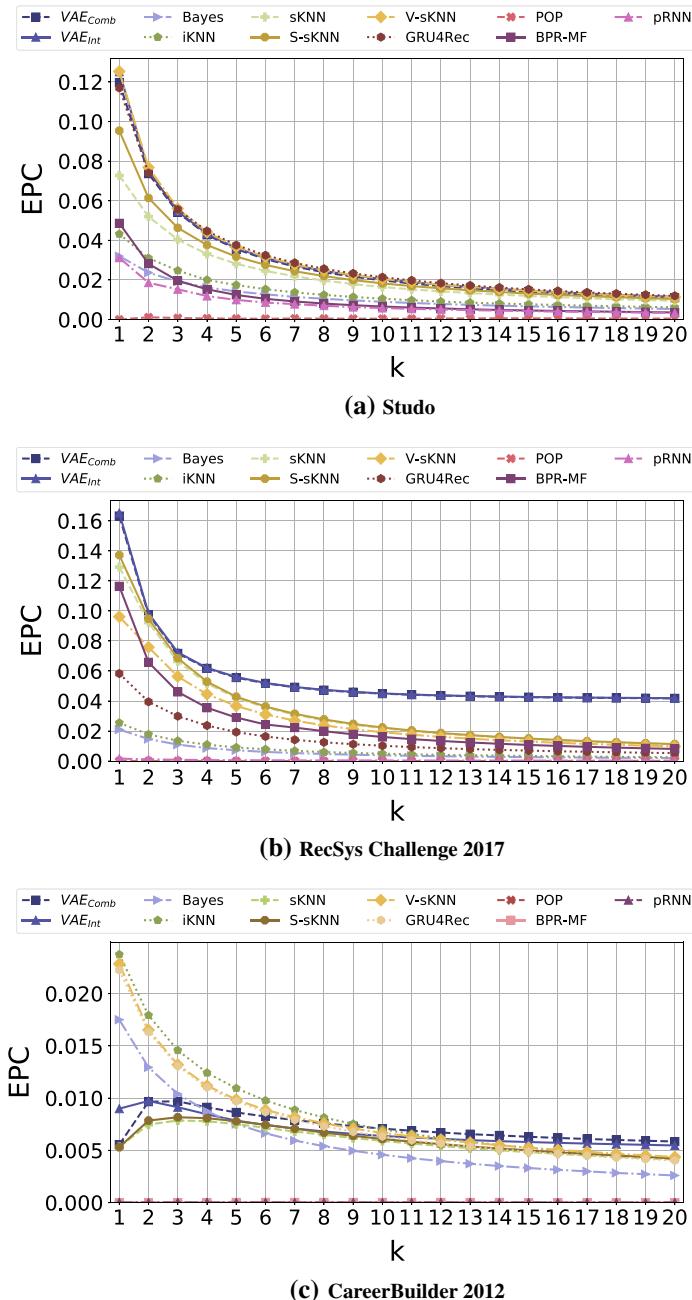


Fig. 9 EPC results for different recommendation list sizes (i.e., values of k) when predicting the next job in the session. On all three datasets, both our proposed VAE approaches achieve good results concerning beyond-accuracy (i.e., EPC) metrics

notable differences among the datasets. First, the *Bayes* approach establishes itself as a competitive baseline in the Studo dataset, whereas it results in a poor performance for the two larger datasets (i.e., RecSys17 and CareerBuilder12). In fact, for the RecSys17 dataset, it results in the worst performance. Hence, when the domain has a small number of items, it can be reasonable to employ such a simple and computationally inexpensive method.

Second, the accuracy of *POP* in the RecSys17 dataset is noteworthy.¹⁴ The reason for this is that in the RecSys17 dataset, the most popular job from the train set was also the one with the highest number of interactions in the test set (i.e., around 21.5%). However, this approach will likely not result in high user satisfaction, just by predicting the same items repeatedly. Moreover, the *BPR-MF* performs best in terms of nDCG in the RecSys17 dataset, but it has the second worst performance in the other two datasets. Also, *GRU4Rec* performs worse for the RecSys17 dataset when compared to Studo and CareerBuilder12. We attribute this to bias toward popularity (Ludewig and Jannach 2018). The performance of *GRU4Rec* is low, while the performance of *BPR-MF* is high in the RecSys17 dataset. The *pRNN* method performs low on all three datasets, but its recommendation accuracy is especially weak on the CareerBuilder12 dataset. Finally, the performance of the *iKNN* differs among all three datasets. While it has the highest MRR for the CareerBuilder12 dataset, the performance in the RecSys17 dataset is the second lowest for both accuracy metrics.

For the next job prediction problem shown in Fig. 8, in all three datasets, all approaches show a similar accuracy performance. The results confirm the presence of bias toward popular items in the RecSys17 dataset as the popularity approach outperforms the other algorithms until $k = 3$, after which *BPR-MF* becomes the best performing approach. We also attribute the sudden increase in the nDCG values for *BPR-MF* and *pRNN* at the recommendation list of length 4 to this popularity bias in the dataset. A closer inspection revealed that both approaches often recommend highly popular items from the train set at the beginning of the recommendation list. The top-1 (i.e., most popular) item that is shared between the train and test set is also the one which gets recommended most frequently as the fourth item in the recommendation list of *BPR-MF* and *pRNN*. Besides that, for all values of k (i.e., the number of recommended jobs), the session-based KNN approaches and *GRU4Rec* achieve competitive accuracy values.

System-based novelty (EPC) As shown in Table 4, both VAE approaches achieve top results in terms of EPC for all three datasets. VAE_{Int} performs best on the RecSys17 dataset, while VAE_{Comb} outperforms all approaches in the CareerBuilder12 dataset. In the Studo dataset, VAE_{Int} achieves second best to *GRU4Rec*. Especially in the RecSys17 dataset, the difference in novelty is considerably high when compared to other baselines. For the baselines, the *sKNN* approaches and *GRU4Rec* both exhibit a good performance concerning the novelty of the recommended jobs. The *pRNN* method, as well as *POP* and *BPR-MF*, produces recommendations that have the lowest system-based novelty.

¹⁴ Quadrana et al. (2017) report that their popularity approach outperforms session-based RNN (Hidasi et al. 2015) in the XING dataset used in the ACM RecSys Challenge 2016.

Table 5 Summary of the rankings of the session-based algorithms evaluated in the job domain. (Color table online)

	Accuracy	Beyond Accuracy	Coverage
VAE _{Int}	++	++	++
VAE _{Comb}	+	++	++
sKNN	+	o	+
V-sKNN	++	+	++
S-sKNN	++	+	+
GRU4Rec	++	+	+
pRNN	--	--	--
Bayes	--	--	o
iKNN	o	-	+
BPR-MF	-	--	++
POP	--	--	--

“++” indicates best, “+” good, “o” average, “-” low and “--” the worst ranking with respect to (1) accuracy (i.e., nDCG and MRR), (2) beyond-accuracy (i.e., EPC and EPD) and (3) coverage

In Fig. 9, we see that both our proposed VAE approaches outperform all others in the CareerBuilder12 dataset after $k = 9$. The sKNN baselines, as well as GRU4Rec, show a better novelty performance for a smaller number of recommended jobs.

Session-based novelty (EPD) As depicted in Table 4, both VAE approaches provide the best session-based novelty for the RecSys17 and CareerBuilder12 datasets and are competitive in the Studio dataset. The VAE_{Comb} method generates the most surprising recommendations in the largest dataset (i.e., CareerBuilder12) and GRU4Rec in the smallest dataset (i.e., Studio). In all cases, the sKNN-based approaches are a competitive baseline. We can observe the most notable difference between accuracy and EPD, however, in the CareerBuilder12 dataset, where the VAE approaches result in a rather average accuracy while performing very well concerning session-based novelty. Overall, the results indicate that both VAE approaches are suitable for cases when we aim to generate novel session-based recommendations.

Coverage In Table 4, we report the percentage of jobs, which were recommended and are a part of (1) all jobs available in the dataset (i.e., the complete item catalog), and (2) the jobs that we know anonymous session users will interact within the test set (i.e., the expected item catalog).

In terms of the coverage of all possible job postings, VAE_{Comb} performs best in the Studio dataset. BPR-MF covers at the most the entire item catalog in the RecSys17 and CareerBuilder12 dataset. Concerning the coverage of items in the test set (i.e., expected items), the session-based KNN approaches achieve almost perfect coverage in the Studio dataset. Only in the case of the RecSys17 dataset, the

BPR-MF baseline has an even higher coverage. As expected, the *POP* baseline results in the worst coverage. While this baseline has high accuracy values in the RecSys17 dataset (due to the popularity bias inherent in this dataset), it effectively covers only a small fraction of jobs in the system. It also has to be noted that the *pRNN* baseline always has the second-worst coverage. As the available item catalog grows, the coverage drops, which suggests that the trained model focuses on a specific (i.e., relatively small) set of items, which explains the worse performance in the largest dataset (i.e., CareerBuilder12).

5.5 Performance overview

To provide a better overview of the performance of the different session-based job recommendation approaches, we summarize all results in Table 5 with respect to three metric categories. That is, we report the performance on accuracy (i.e., *nDCG* and *MRR*), beyond accuracy (i.e., *EPC* and *EPD*) and coverage (of the whole dataset and the test set). For every approach, we assign a rank (i.e., from 1 to 11) for the particular metric in a dataset. We then aggregate these rankings across all three metric categories and datasets. The final rankings are then normalized and assigned into five performance buckets (i.e., from worst “- -” to best “++”; see “Appendix C” for the calculation steps).

Concerning accuracy, the best performance is achieved by *V-sKNN*, our VAE_{Int} variant, *S-sKNN* and *GRU4Rec*. This is then followed by VAE_{Comb} and *sKNN*. All other baselines achieve worse accuracy. For the beyond accuracy metric category, both of our *VAE* variants achieve the best performance. This is followed by *GRU4Rec* and the *sKNN* variants. A similar observation can be made for the metric category coverage. Here, however, *BPR-MF* also shows the best, *iKNN* good and the simple *Bayes* baseline medium coverage. Noteworthy is also the ranking score of the VAE_{Comb} , as with our proposed method it is possible to train the autoencoder models faster (i.e., even with a large item space) and without the need to frequently retrain the utilized model to consider new jobs coming to the system. The *pRNN* approach did not achieve a good rank in any metric category. The same is true for *POP*.

6 Conclusion and future work

In this work, we addressed the problem of providing job recommendations in an anonymous, online session setting. In three datasets, i.e., Studio, RecSys17 and CareerBuilder12, we evaluated the efficacy of using different autoencoder architectures to produce session-based job recommendations. Specifically, we utilized autoencoders to infer latent session representations, which are used in a k-nearest neighbor manner to recommend jobs within a session. We evaluated two types of

input for the autoencoders: (1) interactions with job postings within browsing sessions and (2) a combination of interactions with job postings and content features extracted from these job postings.

We found that variational autoencoders trained on interaction and content data, and used in a k -nearest neighbor manner, led to very good results in terms of accuracy compared to other autoencoder variants. A visual analysis of the embedding spaces with t-SNE revealed that we could attribute a lower accuracy performance when similar-sized sessions form clusters in the 2D space. Although this was mostly the case for autoencoders trained on content features, in practice, however, such an approach has the advantage of fixed size vectors, which means retraining is needed less often. Consequently, depending on the application scenario, one can decide which input for the variational autoencoder to take, i.e., to balance frequent retraining and accuracy.

Furthermore, we evaluated all autoencoder and baseline approaches with respect to beyond-accuracy metrics, i.e., system-based and session-based novelty as well as coverage, in two settings: Firstly, we compared the recommendation performance of the approaches on all remaining interactions within a session, and secondly, we predicted the next job interaction in the session. We find that our proposed variational autoencoder methods can outperform state-of-the-art approaches for sessions-based recommender systems with respect to system-based and session-based novelty. Besides, the session-based KNN approaches are a competitive baseline for the variational autoencoder methods with respect to accuracy and coverage.

For future work, we aim to explore the use of generative variational autoencoder models to directly recommend jobs from the reconstructed session vector (e.g., in a similar way as in Liang et al. 2018). Other ideas for future work include investigating all approaches used in this study in an online evaluation. We plan to conduct an online study to ask users how satisfied and surprised they are with job recommendations generated by autoencoders. Also, we plan to evaluate the accuracy in an A/B test to conclude whether a higher system-based and session-based novelty in a session-based offline setting leads to higher user satisfaction. Additionally, we also plan to directly optimize for the beyond-accuracy metrics by incorporating re-ranking techniques (e.g., maximum marginal relevance Carbonell and Goldstein 1998). These evaluations are planned to be carried out in the Talto¹⁵ career platform. In summary, we hope that the approach presented in this paper will attract further research on the effectiveness of dimensionality reduction techniques.

¹⁵ Talto (<https://talto.com>) is the successor of the jobs platform in Studo (<http://www.studo>).

for session-based job recommender systems and the effect of such methods on beyond-accuracy metrics such as system-based and session-based novelty as well as coverage.

Limitations Our work has several limitations. So far, we only focused on autoencoders to infer the latent representation of the anonymous user session. While autoencoders are a popular choice to reduce the dimensionality of data, other deep neural networks such as restricted Boltzmann machines (Nguyen et al. 2013), deep belief networks (Srivastava and Salakhutdinov 2012) or convolutional neural networks (Shen et al. 2014) could also serve well for this task. Furthermore, additional metadata information about jobs (e.g., textual content of job postings) could potentially enhance recommendations, which we did not tackle due to the unavailability of such data in all datasets. So far, we did not compare the approaches used in this study concerning computational performance, like the authors of (Ludewig and Jannach 2018) did. Moreover, in this work, we did not investigate how to model repeated interactions on the same job postings. Although this is implicitly considered by the autoencoder variants that combine interactions with job content features, such actions are not taken into account by the autoencoders that solely rely on interaction data. Also, in this work, we extracted the candidate sessions based on the last job interaction, which is a limitation of our work. For the evaluation procedure, we used a single time-based split for our experiments. One approach to assess the robustness of our results would be to apply a sliding window approach to generate splits with varying lengths. However, the size of the Studio and RecSys17 datasets is limited, which makes such an approach infeasible. For the larger CareerBuilder dataset, a sliding-window-based evaluation approach could be applied to test the robustness of the method. Due to computational constraints, for the present work, we used the same time-based split as for the Studio and the RecSys17 datasets. We leave the exploration of more splits to future work.

Another limitation is that one of the datasets we used for our study, the Studio dataset, is proprietary, and due to the terms of service of Moshbit, the owner of Studio, it cannot be made available for others at this point.

Acknowledgements Open access funding provided by Graz University of Technology. This work is supported by the Know-Center, the Institute of Interactive Systems and Data Science (ISDS) of Graz University of Technology and Moshbit. We thank Moshbit for granting access to their dataset in Studio Jobs. We thank Simone Kopeinik, Dieter Theiler, Tomislav Duricic and Leon Fadjevic for their feedback on this manuscript.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

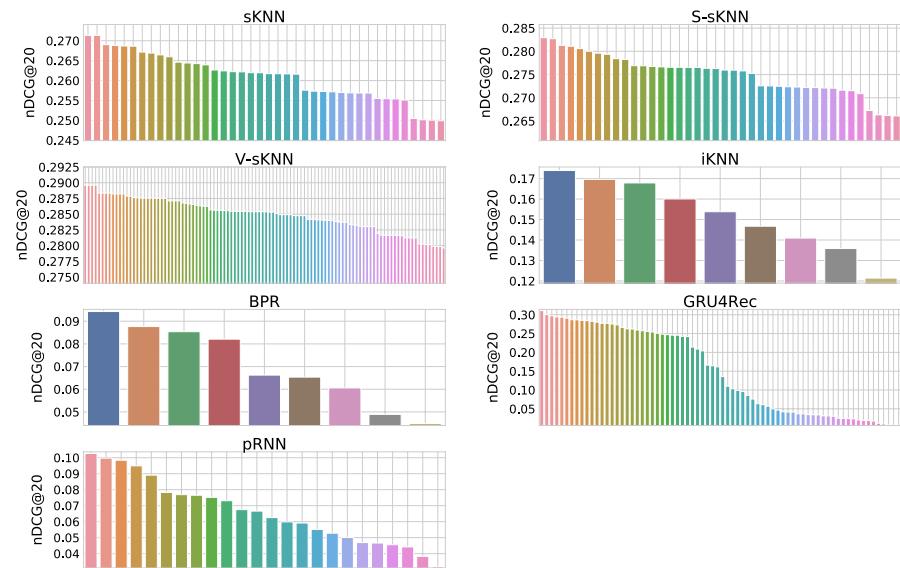


Fig. 10 Accuracy results for the different hyperparameters of the baseline approaches on the Studio dataset

Appendices

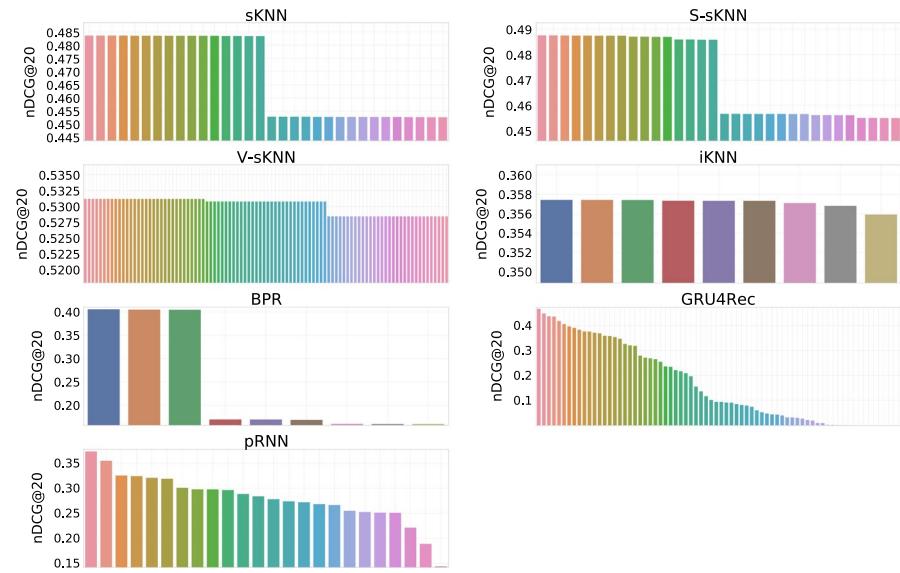


Fig. 11 Accuracy results for the different hyperparameters of the baseline approaches on the RecSys17 dataset

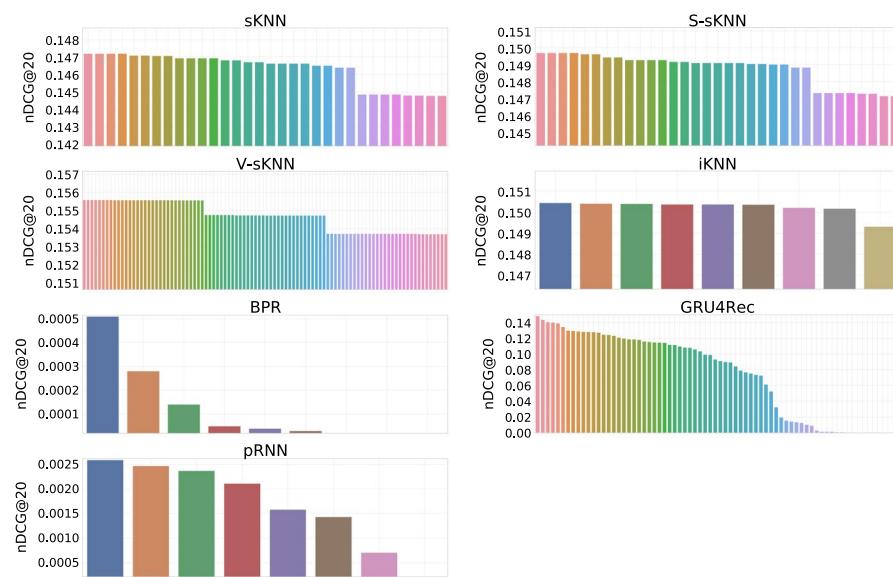


Fig. 12 Accuracy results for the different hyperparameters of the baseline approaches on the CareerBuilder12 dataset

Hyperparameter optimization results

In this section, we report the distribution of the accuracy results achieved by optimizing the hyperparameters for the baseline approaches in Sect. 5.3. For each baseline approach, we pick those hyperparameters which showed the best performance with respect to nDCG@20. As such, Fig. 10 shows the differences between the evaluated baseline configurations on the Studio dataset. Respectively, Fig. 11 depicts the results for the RecSys17 and Fig. 12 for the CareerBuilder12 dataset.

Autoencoder embedding analysis

Figure 13 shows all autoencoder models as t-SNE plots for the Studio dataset, i.e., AE_{Int} , DAE_{Int} and VAE_{Int} trained on interactions and AE_{Comb} , DAE_{Comb} and VAE_{Comb} trained on the combination of interactions and job content. The same is reported for RecSys17 in Fig. 14 and CareerBuilder12 in Fig. 15.

The results indicate that both denoising autoencoders and variational autoencoders tend to produce more session clusters than a classic autoencoder, which creates more of a linear pattern of neighboring sessions. In some cases, we can observe that both the classic and denoising autoencoder models produce shapes without clear structure and large dispersion (e.g., see Fig. 13d or 15b), which indicates that it is hard to find a clear neighborhood of similar sessions. For the smaller Studio dataset, if the autoencoders are solely trained on interactions, i.e., AE_{Int} , DAE_{Int} and VAE_{Int} , more clusters are produced with sessions of different

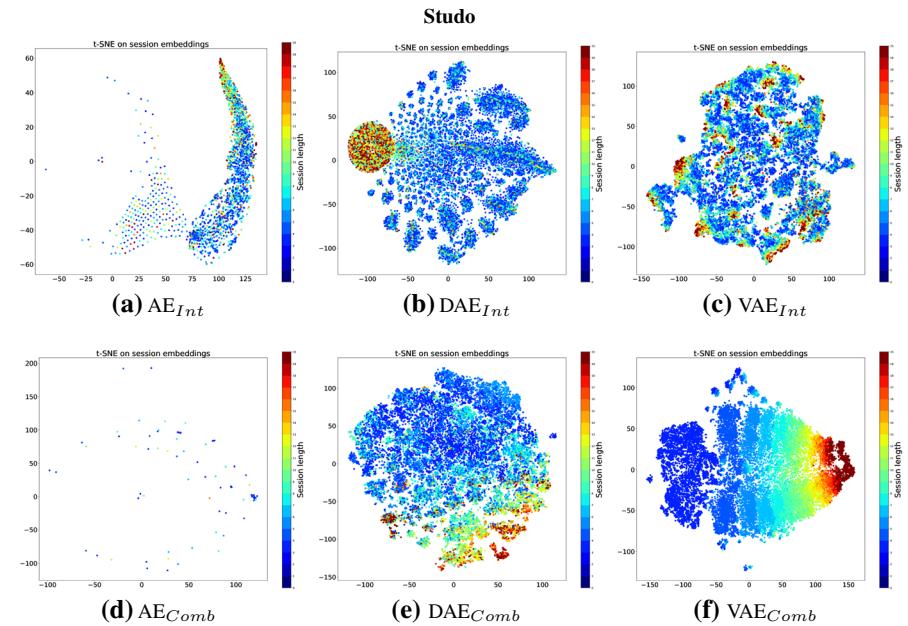


Fig. 13 t-SNE embeddings for latent session representations produced with the three autoencoder models trained on interaction and content data from the Studo dataset. Sessions are colored according to their length, where the same red color is used for sessions with 20 or more interactions

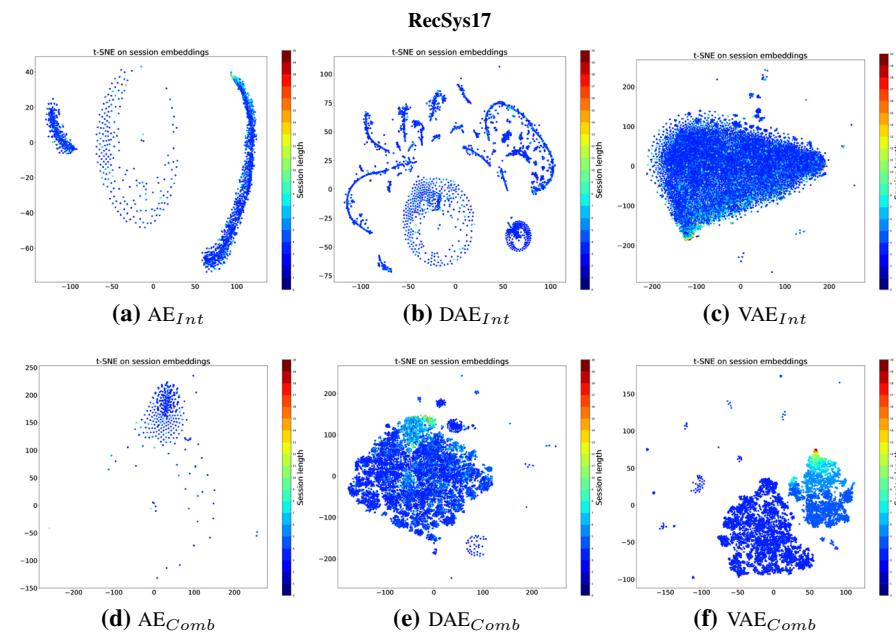


Fig. 14 t-SNE embeddings for latent session representations for the RecSys17 dataset

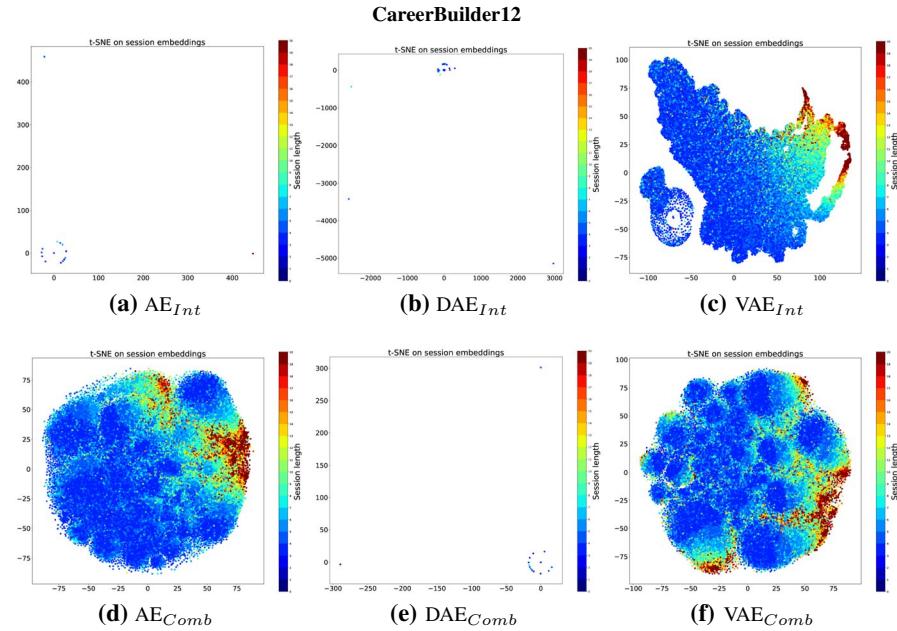


Fig. 15 t-SNE embeddings for latent session representations for the Careerbuilder12 dataset

sizes close to each other (e.g., Fig. 13b, c). Interestingly, if autoencoders are trained on content, we can observe rainbow-colored shapes that are based on session length (e.g., as shown in Fig. 13e, f). In case of a larger dataset like CareerBuilder12, we end up with several sub-clusters that exhibit this rainbow pattern. This shows that when we encode the input with content features, sessions of similar length tend to cluster. We attribute this to sessions of similar length having similar patterns of input vectors (e.g., many right-padded zeros for short sessions).

Aggregation of rankings

In Sect. 5.5, we report the aggregated performance of the different approaches. For this, in Table 6 we first rank the results from each dataset (i.e., based on Table 4). We then sum the rankings for each dataset (i.e., Studio, RecSys17 and CareerBuilder12) for the accuracy metrics (i.e., nDCG and MRR), the beyond-accuracy metrics (i.e., EPC and EPD) and both coverage, respectively. The aggregated rankings are outlined in Table 7. The rankings are then normalized with the equation

Table 6 Ranking of the results per metric and dataset, which are derived from numerical results. (Color table online)

Studo						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	3	3	2	3	3	1
VAE _{Comb}	5	4	5	6	1	1
sKNN	6	6	6	5	7	5
V-sKNN	2	2	3	2	5	1
S-sKNN	4	5	4	4	6	5
GRU4Rec	1	1	1	1	8	8
pRNN	9	9	9	9	10	10
Bayes	8	8	8	8	2	1
iKNN	7	7	7	7	4	5
BPR-MF	10	10	10	11	9	9
POP	11	11	11	10	11	11
RecSys Challenge 2017						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	2	2	1	1	7	4
VAE _{Comb}	3	3	2	2	6	5
sKNN	6	6	4	5	4	3
V-sKNN	7	7	5	5	5	6
S-sKNN	5	5	3	4	3	2
GRU4Rec	8	8	7	8	2	7
pRNN	9	9	10	10	10	10
Bayes	11	11	9	11	9	9
iKNN	10	10	8	8	8	8
BPR-MF	1	4	6	7	1	1
POP	4	1	11	3	11	11
CareerBuilder 2012						
	nDCG	MRR	EPC	EPD	Coverage (%)	
VAE _{Int}	7	7	2	2	3	1
VAE _{Comb}	6	6	1	1	6	2
sKNN	4	5	7	6	8	6
V-sKNN	1	2	3	4	5	3
S-sKNN	2	4	5	5	7	5
GRU4Rec	3	3	4	3	2	8
pRNN	9	9	9	9	10	10
Bayes	8	8	8	8	9	9
iKNN	5	1	6	7	4	7
BPR-MF	9	10	9	9	1	4
POP	11	10	9	9	11	11

Coloring is according to the rank within each dataset

Table 7 Aggregated rankings across the three different datasets and per metric type (i.e., accuracy, beyond accuracy and coverage)

	Accuracy		Beyond Accuracy		Coverage	
	Aggregated	Normalized	Aggregated	Normalized	Aggregated	Normalized
VAE _{Int}	24	0.1176	11	0.0217	19	0.0208
VAE _{Comb}	27	0.2059	17	0.1522	21	0.0625
sKNN	33	0.3824	33	0.5000	33	0.3125
V-sKNN	21	0.0294	22	0.2609	25	0.1458
S-sKNN	25	0.1471	25	0.3261	28	0.2083
GRU4Rec	24	0.1176	24	0.3043	35	0.3542
pRNN	54	1.0000	56	1.0000	60	0.8750
Bayes	54	1.0000	52	0.9130	39	0.4375
iKNN	40	0.5882	43	0.7174	36	0.3750
BPR-MF	44	0.7059	52	0.9130	25	0.1458
POP	48	0.8235	53	0.9348	66	1.0000

Results are then normalized by a min-max scaling

$Norm(x) = \frac{x-min+1}{max-min+1}$, where min is the lowest aggregated rank and max is the highest aggregated rank. Thus, lower results are considered better, while the worst results receive the value 1. The results are then put into five buckets according to their values. A double plus (i.e.,++) is assigned to values between 0.0 and 0.2, while values between 0.2 and 0.4 get assigned a single plus (i.e., +), followed by o (i.e., 0.4 until 0.6), – (i.e., 0.6 until 0.8) and for the worst results a -- (i.e., 0.8 until 1).

References

- Abel, F.: We know where you should work next summer: job recommendations. In: Proceedings of the 9th ACM Conference on Recommender Systems, pp. 230–230 (2015)
- Abel, F., Benczúr, A., Kohlsdorf, D., Larson, M., Pálovics, R.: Recsys challenge 2016: job recommendations. In: Proceedings of the 10th ACM Conference on Recommender Systems, pp. 425–426. ACM (2016)
- Abel, F., Deldjoo, Y., Elahi, M., Kohlsdorf, D.: Recsys challenge 2017: offline and online evaluation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 372–373 (2017)
- Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans. Knowl. Data Eng. **24**(5), 896–911 (2012)
- Aggarwal, C.C.: Evaluating recommender systems. In: Recommender Systems, pp. 225–254. Springer (2016)
- Al-Otaibi, S.T., Ykhlef, M.: A survey of job recommender systems. Int. J. Phys. Sci. **7**(29), 5127–5142 (2012)
- Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: Advances in Neural Information Processing Systems, pp. 153–160 (2007)
- Bianchi, M., Cesaro, F., Ciciri, F., Dagrada, M., Gasparin, A., Grattarola, D., Inajjar, I., Metelli, A.M., Celli, L.: Content-based approaches for cold-start job recommendations. In: Proceedings of the Recommender Systems Challenge 2017, p. 6. ACM (2017)
- Bonnin, G., Jannach, D.: Automated generation of music playlists: survey and experiments. ACM Comput. Surv. (CSUR) **47**(2), 26 (2015)
- Carbonell, J., Goldstein, J.: The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 335–336 (1998)

- Chatzis, S.P., Christodoulou, P., Andreou, A.S.: Recurrent latent variable networks for session-based recommendation. In: Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems, pp. 38–45. ACM (2017)
- Fischer, A., Igel, C.: An introduction to restricted Boltzmann machines. In: Iberoamerican Congress on Pattern Recognition, pp. 14–36. Springer (2012)
- He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th International Conference on World Wide Web, pp. 173–182. International World Wide Web Conferences Steering Committee (2017)
- Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst. (TOIS)* **22**(1), 5–53 (2004)
- Hidasi, B., Karatzoglou, A.: Recurrent neural networks with top-k gains for session-based recommendations. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 843–852. ACM (2018)
- Hidasi, B., Karatzoglou, A., Baltrunas, L., Tikk, D.: Session-based recommendations with recurrent neural networks. arXiv preprint [arXiv:1511.06939](https://arxiv.org/abs/1511.06939) (2015)
- Hidasi, B., Quadrana, M., Karatzoglou, A., Tikk, D.: Parallel recurrent neural network architectures for feature-rich session-based recommendations. In: Proceedings of the 10th ACM Conference on Recommender Systems, pp. 241–248. ACM (2016)
- Hidasi, B., Tikk, D.: General factorization framework for context-aware recommendations. *Data Min. Knowl. Discov.* **30**(2), 342–371 (2016)
- Hong, W., Zheng, S., Wang, H., Shi, J.: A job recommender system based on user clustering. *JCP* **8**(8), 1960–1967 (2013)
- Jannach, D., Ludewig, M.: When recurrent neural networks meet the neighborhood for session-based recommendation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 306–310 (2017)
- Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. *Mach. Learn.* **37**(2), 183–233 (1999)
- Kamehkhosh, I., Jannach, D., Ludewig, M.: A comparison of frequent pattern techniques and a deep learning method for session-based recommendation. In: RecTemp@ RecSys, pp. 50–56 (2017)
- Kenthapadi, K., Le, B., Venkataraman, G.: Personalized job recommendation system at linkedin: practical challenges and lessons learned. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 346–347 (2017)
- Kenthapadi, K., Le, B., Venkataraman, G.: Personalized job recommendation system at linkedin: practical challenges and lessons learned. In: Proceedings of the 11th ACM Conference on Recommender Systems, pp. 346–347 (2017)
- Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint. [arXiv:1312.6114](https://arxiv.org/abs/1312.6114) (2013)
- Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer* **42**(8), 30–37 (2009)
- Kramer, M.A.: Nonlinear principal component analysis using autoassociative neural networks. *AIChE J.* **37**(2), 233–243 (1991)
- Lacic, E., Kowald, D., Traub, M., Luzhnica, G., Simon, J., Lex, E.: Tackling cold-start users in recommender systems with indoor positioning systems. In: Poster Proceedings of the 9th ACM Conference on Recommender Systems (2015)
- Lacic, E., Reiter-Haas, M., Duricic, T., Slawicek, V., Lex, E.: Should we embed? A study on the online performance of utilizing embeddings for real-time job recommendations. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp. 496–500. ACM (2019)
- Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T., Ma, J.: Neural attentive session-based recommendation. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1419–1428. ACM (2017)
- Liang, D., Krishnan, R.G., Hoffman, M.D., Jebara, T.: Variational autoencoders for collaborative filtering. arXiv preprint. [arXiv:1802.05814](https://arxiv.org/abs/1802.05814) (2018)
- Lin, Z., Feng, M., Santos, C.N.d., Yu, M., Xiang, B., Zhou, B., Bengio, Y.: A structured self-attentive sentence embedding. arXiv preprint. [arXiv:1703.03130](https://arxiv.org/abs/1703.03130) (2017)
- Liu, Q., Zeng, Y., Mokhosi, R., Zhang, H.: Stamp: short-term attention/memory priority model for session-based recommendation. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1831–1839. ACM (2018)
- Liu, R., Rong, W., Ouyang, Y., Xiong, Z.: A hierarchical similarity based job recommendation service framework for university students. *Front. Comput. Sci.* **11**(5), 912–922 (2017)

- Ludewig, M., Jannach, D.: Evaluation of session-based recommendation algorithms. *User Model. User-Adap. Inter.* **28**(4–5), 331–390 (2018)
- Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. *J. Mach. Learn. Res.* **9**(Nov), 2579–2605 (2008)
- Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint. [arXiv:1511.05644](https://arxiv.org/abs/1511.05644) (2015)
- Matuszyk, P., Vinagre, J., Spiliopoulou, M., Jorge, A.M., Gama, J.: Forgetting methods for incremental matrix factorization in recommender systems. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing, pp. 947–953. ACM (2015)
- McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: ACM CHI'06 (2006)
- Mine, T., Kakuta, T., Ono, A.: Reciprocal recommendation for job matching with bidirectional feedback. In: 2013 Second IIAI International Conference on Advanced Applied Informatics, pp. 39–44. IEEE (2013)
- Mishra, S.K., Reddy, M.: A bottom-up approach to job recommendation system. In: Proceedings of the Recommender Systems Challenge, p. 4. ACM (2016)
- Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on International Conference on Machine Learning, pp. 807–814 (2010)
- Nguyen, T.D., Tran, T., Phung, D., Venkatesh, S.: Learning sparse latent representation and distance metric for image retrieval. In: 2013 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2013)
- Parikh, A., Täckström, O., Das, D., Uszkoreit, J.: A decomposable attention model for natural language inference. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2249–2255 (2016)
- Parra, D., Sahebi, S.: Recommender systems: sources of knowledge and evaluation metrics. In: Advanced Techniques in Web Intelligence-2: Web User Browsing Behaviour and Preference Analysis, pp. 149–175. Springer (2013)
- Pu, P., Chen, L., Hu, R.: A user-centric evaluation framework for recommender systems. In: Proceedings of the fifth ACM conference on Recommender systems, pp. 157–164 (2011)
- Quadrana, M., Karatzoglou, A., Hidasi, B., Cremonesi, P.: Personalizing session-based recommendations with hierarchical recurrent neural networks. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 130–137. ACM (2017)
- Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
- Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. arXiv preprint. [arXiv:1401.4082](https://arxiv.org/abs/1401.4082) (2014)
- Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM (2001)
- Sedhain, S., Menon, A.K., Sanner, S., Xie, L.: Autorec: autoencoders meet collaborative filtering. In: Proceedings of the 24th International Conference on World Wide Web, pp. 111–112. ACM (2015)
- Shani, G., Heckerman, D., Brafman, R.I.: An mdp-based recommender system. *J. Mach. Learn. Res.* **6**(Sep), 1265–1295 (2005)
- Shen, Y., He, X., Gao, J., Deng, L., Mesnil, G.: A latent semantic model with convolutional-pooling structure for information retrieval. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, pp. 101–110. ACM (2014)
- Siting, Z., Wenxing, H., Ning, Z., Fan, Y.: Job recommender systems: a survey. In: 2012 7th International Conference on Computer Science Education (ICCSE), pp. 920–924 (2012)
- Smirnova, E., Vasile, F.: Contextual sequence modeling for recommendation with recurrent neural networks. arXiv preprint. [arXiv:1706.07684](https://arxiv.org/abs/1706.07684) (2017)
- Song, Y., Elkahky, A.M., He, X.: Multi-rate deep learning for temporal recommendation. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 909–912. ACM (2016)
- Srivastava, N., Salakhutdinov, R.: Learning representations for multimodal data with deep belief nets. In: International Conference on Machine Learning Workshop, Vol. 79 (2012)
- Strub, F., Gaudel, R., Mary, J.: Hybrid recommender system based on autoencoders. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp. 11–16. ACM (2016)

- Tan, Y.K., Xu, X., Liu, Y.: Improved recurrent neural networks for session-based recommendations. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp. 17–22 (2016)
- Theis, L., Shi, W., Cunningham, A., Huszár, F.: Lossy image compression with compressive autoencoders. arXiv preprint. [arXiv:1703.00395](https://arxiv.org/abs/1703.00395) (2017)
- Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop, coursera: neural networks for machine learning. University of Toronto, Technical Report (2012)
- Tuan, T.X., Phuong, T.M.: 3d convolutional networks for session-based recommendation with content features. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 138–146. ACM (2017)
- Twardowski, B.: Modelling contextual information in session-aware recommender systems with neural networks. In: Proceedings of the 10th ACM Conference on Recommender Systems, pp. 273–276. ACM (2016)
- Vargas, S., Castells, P.: Rank and relevance in novelty and diversity metrics for recommender systems. In: Proceedings of the Fifth ACM Conference on Recommender Systems, pp. 109–116. ACM (2011)
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
- Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning, pp. 1096–1103. ACM (2008)
- Volkovs, M., Yu, G.W., Poutanen, T.: Content-based neighbor models for cold start in recommender systems. In: Proceedings of the Recommender Systems Challenge 2017, p. 7. ACM (2017)
- Voorhees, E.: Proceedings of the 8th Text Retrieval Conference. TREC-8 Question Answering Track Report, pp. 77–82 (1999)
- Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., Tan, T.: Session-based recommendation with graph neural networks. arXiv preprint. [arXiv:1811.00855](https://arxiv.org/abs/1811.00855) (2018)
- Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., Tan, T.: Session-based recommendation with graph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, pp. 346–353 (2019)
- Wu, Y., DuBois, C., Zheng, A.X., Ester, M.: Collaborative denoising auto-encoders for top-n recommender systems. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, pp. 153–162. ACM (2016)
- Xiao, W., Xu, X., Liang, K., Mao, J., Wang, J.: Job recommendation with hawkes process: an effective solution for recsys challenge 2016. In: Proceedings of the Recommender Systems Challenge, p. 11. ACM (2016)
- Yuan, F., Karatzoglou, A., Arapakis, I., Jose, J.M., He, X.: A simple convolutional generative network for next item recommendation. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 582–590 (2019)
- Zhang, C., Cheng, X.: An ensemble method for job recommender systems. In: Proceedings of the Recommender Systems Challenge, p. 2. ACM (2016)
- Zhang, Y.C., Séaghdha, D.O., Quercia, D., Jambor, T.: Auralist: introducing serendipity into music recommendation. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, pp. 13–22. ACM (2012)
- Zhou, T., Kuscsik, Z., Liu, J.G., Medo, M., Wakeling, J.R., Zhang, Y.C.: Solving the apparent diversity-accuracy dilemma of recommender systems. Proc. Natl. Acad. Sci. **107**(10), 4511–4515 (2010)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Emanuel Lacic is a Senior Researcher and Recommender Systems Architect in the Social Computing team at the Know-Center. He is a PhD student at Graz University of Technology and a former visiting researcher at the Computer Science department of the University of California, Los Angeles. He has an M.Sc. and B.Sc. in Software Engineering and Information Systems from the University of Zagreb. His research interests are in the fields of Recommender Systems, Deep Learning and Social Network Analysis.

Markus Reiter-Haas is a researcher at Moshbit GmbH and is responsible for the recommender system of the Talto career platform. He has a background in Computer Science at the Graz University of Technology with a focus on Knowledge Technologies. His master thesis tackled the evaluation of student job recommendations on the Talto predecessor Studio Jobs. His current research concentrates on creating low-dimensional embeddings for effective retrieval in the job domain.

Dominik Kowald is a post-doctoral researcher and deputy research area manager of the Social Computing team at the Know-Center. He has a Ph.D. (with honours), M.Sc. (with honours) and B.Sc. in Computer Science from Graz University of Technology. His research interests are in the fields of recommender systems, fairness and biases in algorithms, and computational social science, in which he has published more than 60 papers so far.

Manoj Reddy Daredy is a Ph.D. Candidate in the Computer Science department at the University of California Los Angeles. His research interest is in recommender systems and applied machine learning. More specifically, Manoj works on emerging frontiers in personalization such as privacy and explainability. He received his Masters from the University of Michigan Ann Arbor and Bachelors from Carnegie Mellon University in Qatar.

Junghoo Cho is a professor in the Department of Computer Science at the University of California, Los Angeles. He received a Ph.D. degree in Computer Science from Stanford University and a B.S. degree in physics from Seoul National University. His research interest is in the theory and practice of learning, particularly in the area of language acquisition and understanding. He is a recipient of prestigious awards such as the 10-Year Best Paper Award at VLDB 2010, NSF CAREER Award or IBM Faculty Award.

Elisabeth Lex is an assistant professor and head of the Social Computing Lab at Graz University of Technology, Austria. She received a Ph.D. and an M.Sc. degree in Computer Science from Graz University of Technology. Her research interests are in the development of personalized recommender systems, in particular, algorithms based on psychological theory, as well as in computational social science, more specifically, using behavioral and network data to investigate human activity and social dynamics.

Affiliations

Emanuel Lacic¹ · Markus Reiter-Haas² · Dominik Kowald¹ ·
Manoj Reddy Daredy³ · Junghoo Cho³ · Elisabeth Lex⁴ 

Emanuel Lacic
elacic@know-center.at

Markus Reiter-Haas
markus.reiter-haas@moshbit.com

Dominik Kowald
dkowald@know-center.at

Manoj Reddy Daredy
mdaredy@cs.ucla.edu

Junghoo Cho
cho@cs.ucla.edu

¹ Know-Center GmbH, Graz, Austria

² Moshbit GmbH, Graz, Austria

³ University of California, Los Angeles, USA

⁴ Graz University of Technology, Graz, Austria

P8 Robustness of Meta Matrix Factorization Against Strict Privacy Constraints (2021)

Privacy and Limited Preference Information in Recommender Systems

[P8] Muellner, P., **Kowald, D.**, Lex, E. (2021). Robustness of Meta Matrix Factorization Against Strict Privacy Constraints. In *Proceedings of the 43rd European Conference on Information Retrieval (ECIR'2021)*, pp. 107-119.
DOI: https://doi.org/10.1007/978-3-030-72240-1_8



Robustness of Meta Matrix Factorization Against Strict Privacy Constraints

Peter Muellner¹, Dominik Kowald¹, and Elisabeth Lex²

¹ Know-Center GmbH, Graz, Austria

{pmuellner,dkowald}@know-center.at

² Graz University of Technology, Graz, Austria

elisabeth.lex@tugraz.at

Abstract. In this paper, we explore the reproducibility of MetaMF, a meta matrix factorization framework introduced by Lin et al. MetaMF employs meta learning for federated rating prediction to preserve users' privacy. We reproduce the experiments of Lin et al. on five datasets, i.e., Douban, Hetrec-MovieLens, MovieLens 1M, Ciao, and Jester. Also, we study the impact of meta learning on the accuracy of MetaMF's recommendations. Furthermore, in our work, we acknowledge that users may have different tolerances for revealing information about themselves. Hence, in a second strand of experiments, we investigate the robustness of MetaMF against strict privacy constraints. Our study illustrates that we can reproduce most of Lin et al.'s results. Plus, we provide strong evidence that meta learning is essential for MetaMF's robustness against strict privacy constraints.

Keywords: Recommender systems · Privacy · Meta learning · Federated learning · Reproducibility · Matrix factorization

1 Introduction

State-of-the-art recommender systems learn a user model from user and item data and the user's interactions with items to generate personalized recommendations. In that process, however, users' personal information may be exposed, resulting in severe privacy threats. As a remedy, recent research makes use of techniques like federated learning [2, 4, 6] or meta learning [7, 20] to ensure privacy in recommender systems. In the federated learning paradigm, no data ever leaves a user's device, and as such, the leakage of their data by other parties is prohibited. With meta learning, a model gains the ability to form its hypothesis based on a minimal amount of data.

Similar to recent work [5, 15], MetaMF by Lin et al. [16] combines federated learning with meta learning to provide personalization and privacy. Besides, MetaMF exploits collaborative information among users and distributes a private rating prediction model to each user. Due to MetaMF's recency and its clear focus on increasing privacy for users via a novel framework, we are interested

in the reproducibility of Lin et al.’s research. Additionally, we aim to contribute our own branch of research regarding privacy, i.e., MetaMF’s robustness against strict privacy constraints. This is motivated by a statement of Lin et al. about one critical limitation of MetaMF, i.e., its sensitivity to data scarcity that could arise when users employ strict privacy constraints by withholding a certain amount of their data. In this regard, every user has a certain privacy budget, i.e., a budget of private data she is willing to share. Thus, in our paper at hand, the privacy budget is considered a measure of how much data disclosure a user tolerates and is defined as the fraction of rating data she is willing to share with others. Thereby, employing small privacy budgets and thus, withholding data, serves as a realization of strict privacy constraints.

Our work addresses MetaMF’s limitation against data scarcity and is structured in two parts. First, we conduct a study with the aim to reproduce the results given in the original work by Lin et al. Concretely, we investigate two leading research questions, i.e., *RQ1a: How does MetaMF perform on a broad body of datasets?* and *RQ1b: What evidence does MetaMF provide for personalization and collaboration?* Second, we present a privacy-focused study, in which we evaluate the impact of MetaMF’s meta learning component and test MetaMF’s performance on users with different amounts of rating data. Here, we investigate two more research questions, i.e., *RQ2a: What is the role of meta learning in the robustness of MetaMF against decreasing privacy budgets?* and *RQ2b: How do limited privacy budgets affect users with different amounts of rating data?* We address *RQ1a* and *RQ1b* in Sect. 3 by testing MetaMF’s predictive capabilities on five different datasets, i.e., Douban, Hetrec-MovieLens, MovieLens 1M, Ciao, and Jester. Here, we find that most results provided by Lin et al. can be reproduced. In Sect. 4, we elaborate on *RQ2a* and *RQ2b* by examining MetaMF in the setting of decreasing privacy budgets. Here, we provide strong evidence of the important role of meta learning in MetaMF’s robustness. Besides, we find that users with large amounts of rating data are substantially disadvantaged by decreasing privacy budgets compared to users with few rating data.

2 Methodology

In this section, we illustrate our methodology of addressing *RQ1a* and *RQ1b*, i.e., the reproducibility of Lin et al. [16], and *RQ2a* and *RQ2b*, i.e., MetaMF’s robustness against decreasing privacy budgets.

2.1 Approach

MetaMF. Lin et al. recently introduced a novel matrix factorization framework in a federated environment leveraging meta learning. Their framework comprises three steps. First, collaborative information among users is collected and subsequently, utilized to construct a user’s collaborative vector. This collaborative vector serves as basis of the second step. Here, in detail, the parameters of

a private rating prediction model are learned via meta learning. Plus, in parallel, personalized item embeddings, representing a user’s personal “opinion” about the items, are computed. Finally, in the third step, the rating of an item is predicted utilizing the previously learned rating prediction model and item embeddings. We resort to MetaMF to address *RQ1a*, *RQ1b*, and *RQ2b*, i.e., the reproducibility of results presented by Lin et al. and the influence of decreasing privacy budgets on users with different amounts of rating data.

NoMetaMF. In our privacy-focused study, *RQ2a* addresses the role of meta learning in MetaMF’s robustness against decreasing privacy budgets. Thus, we conduct experiments with and without MetaMF’s meta learning component. For the latter kind of experiments, we introduce NoMetaMF, a variant of MetaMF with no meta learning. In MetaMF, a private rating prediction model is generated for each user by leveraging meta learning. The authors utilize a hyper-network [11], i.e., a neural network, coined meta network, that generates the parameters of another neural network. Based on the user’s collaborative vector \mathbf{c}_u , the meta network generates the parameters of the rating prediction model, i.e., weights \mathbf{W}_l^u and biases \mathbf{b}_l^u for layer l and user u . This is given by

$$\mathbf{h} = \text{ReLU}(\mathbf{W}_h^* \mathbf{c}_u + \mathbf{b}_h^*) \quad (1)$$

$$\mathbf{W}_l^u = \mathbf{U}_{W_l^u}^* \mathbf{h} + \mathbf{b}_{W_l^u}^* \quad (2)$$

$$\mathbf{b}_l^u = \mathbf{U}_{b_l^u}^* \mathbf{h} + \mathbf{b}_{b_l^u}^* \quad (3)$$

where \mathbf{h} is the hidden state with the widely-used $\text{ReLU}(x) = \max(0, x)$ [8, 12] activation function, \mathbf{W}_h^* , $\mathbf{U}_{W_l^u}^*$, $\mathbf{U}_{b_l^u}^*$ are the weights and \mathbf{b}_h^* , $\mathbf{b}_{W_l^u}^*$, $\mathbf{b}_{b_l^u}^*$ are the biases of the meta network. NoMetaMF excludes meta learning by disabling backpropagation through the meta network in Eqs. 1–3. Thus, meta parameters \mathbf{W}_h^* , $\mathbf{U}_{W_l^u}^*$, $\mathbf{U}_{b_l^u}^*$, \mathbf{b}_h^* , $\mathbf{b}_{W_l^u}^*$, $\mathbf{b}_{b_l^u}^*$ will not be learned in NoMetaMF. While backpropagation is disabled in the meta network, parameters W_l^u and b_l^u are learned over those non-meta parameters in NoMetaMF to obtain the collaborative vector. Hence, the parameters of the rating prediction models are still learned for each user individually, but without meta learning.

Lin et al. also introduce a variant of MetaMF, called MetaMF-SM, which should not be confused with NoMetaMF. In contrast to MetaMF, MetaMF-SM does not generate a private rating prediction model for each user individually, but instead utilizes a shared rating prediction model for all users. Our NoMetaMF model generates an individual rating prediction model for each user but operates without meta learning. Furthermore, we note that in our implementation of NoMetaMF, the item embeddings are generated in the same way as in MetaMF. With NoMetaMF, we aim to investigate the impact of meta learning on the robustness of MetaMF against decreasing privacy budgets, i.e., *RQ2a*.

2.2 Datasets

In line with Lin et al., we conduct experiments on four datasets: Douban [14], Hetrec-MovieLens [3], MovieLens 1M [13], and Ciao [10]. We observe that none

of these datasets comprises a high average number of ratings per item, i.e., 22.6 (Douban), 85.6 (Hetrec-MovieLens), 269.8 (MovieLens 1M), and 2.7 (Ciao). To increase the diversity of our datasets, we include a fifth dataset to our study, i.e., Jester [9] with an average number of ratings per item of 41,363.6. Furthermore, Lin et al. claimed that several observations about Ciao may be explained by its low average number of ratings per user, i.e., 38.3. Since Jester exhibits a similarly low average number of ratings per user, i.e., 56.3, we utilize Jester to verify Lin et al.'s claims. To fit the rating scale of the other datasets, we scale Jester's ratings to a range of [1, 5]. Descriptive statistics of our five datasets are outlined in detail in the following lines. *Douban* comprises 2,509 users with 893,575 ratings for 39,576 items. *Hetrec-MovieLens* includes 10,109 items and 855,598 ratings of 2,113 users. The popular *MovieLens 1M* dataset includes 6,040 users, 3,706 items and 1,000,209 ratings. *Ciao* represents 105,096 items, with 282,619 ratings from 7,373 users. Finally, our additional *Jester* dataset comprises 4,136,360 ratings for 100 items from 73,421 users.

We follow the evaluation protocol of Lin et al. and thus, perform no cross-validation. Therefore, each dataset is randomly separated into 80% training set R_{train} , 10% validation set R_{val} and 10% test set R_{test} . However, we highlight that in the case of Douban, Hetrec-MovieLens, MovieLens 1M, and Ciao, we utilize the training, validation and test set provided by Lin et al.

Identification of User Groups. In *RQ2b*, we study how decreasing privacy budgets influence the recommendation accuracy of user groups with different user behavior. That is motivated by recent research [1, 19], which illustrates differences in recommendation quality for user groups with different characteristics. As an example, [19] measures a user group's mainstreaminess, i.e., how the user groups' most listened artists match the most listened artists of the entire population. The authors split the population into three groups of users with low, medium, and high mainstreaminess, respectively. Their results suggest that low mainstream users receive far worse recommendations than mainstream users.

In a similar vein, we also split users into three user groups: *Low*, *Med*, and *High*, referring to users with a low, medium, and a high number of ratings, respectively. To precisely study the effects of decreasing privacy budgets on each user group, we generate them such that the variance of the number of ratings is low, but yet, include a sufficiently large number of users. For this matter, each of our three user groups includes 5% of all users. In detail, we utilize the 5% of users with the least ratings (i.e., *Low*), the 5% of users with the most ratings (i.e., *High*) and the 5% of users, whose number of ratings are the closest to the median (i.e., *Med*). Thus, each user group consists of 125 (Douban), 106 (Hetrec-MovieLens), 302 (MovieLens 1M), 369 (Ciao), and 3,671 (Jester) users.

2.3 Recommendation Evaluation

In concordance to the methodology of Lin et al., we minimize the mean squared error (MSE) between the predicted $\hat{r} \in \hat{R}$ and the real ratings $r \in R$ as the

objective function for training the model. Additionally, we report the MSE and the mean absolute error (MAE) on the test set R_{test} to estimate our models' predictive capabilities. Since we dedicate parts of this work to shed light on MetaMF's and NoMetaMF's performance in settings with different degrees of privacy, we illustrate how we simulate decreasing privacy budgets and how we evaluate a model's robustness against these privacy constraints.

Simulating Different Privacy Budgets. To simulate the reluctance of users to share their data, we propose a simple sampling procedure in Algorithm 1. Let β be the privacy budget, i.e., the fraction of data to be shared. First, a user u randomly selects a fraction of β of her ratings without replacement. Second, the random selection of ratings R_u^β is then shared by adding it to the set R^β . That ensures that (i) each user has the same privacy budget β and (ii) each user shares at least one rating to receive recommendations. The set of shared ratings R^β without held back ratings then serves as a training set for our models.

Algorithm 1: Sampling procedure for simulating privacy budget β .

Input: Ratings R , Users U and privacy budget β .
Result: Shared ratings R^β , with a fraction of β of each user's ratings.
 $R^\beta = \{\}$
for $u \in U$ **do**
 $R_u^\beta = \{R'_u \subseteq R_u : |R'_u|/|R_u| = \beta\}$
 $R^\beta = R^\beta \cup R_u^\beta$
end

Measuring Robustness. Our privacy-focused study is concerned with discussing MetaMF's robustness against decreasing privacy budgets. We quantify a model's robustness by how the model's predictive capabilities change by decreasing privacy budgets. In detail, we introduce a novel accuracy measurement called $\Delta\text{MAE}@\beta$, which is a simple variant of the mean absolute error.

Definition 1 ($\Delta\text{MAE}@\beta$). *The relative mean absolute error $\Delta\text{MAE}@\beta$ measures the predictive capabilities of a model M under a privacy budget β relative to the predictive capabilities of M without any privacy constraints.*

$$\text{MAE}@\beta = \frac{1}{|R_{test}|} \sum_{r_{u,i} \in R_{test}} |(r_{u,i} - M(R_{train}^\beta, \theta)_{u,i})| \quad (4)$$

$$\Delta\text{MAE}@\beta = \frac{\text{MAE}@\beta}{\text{MAE}@1.0} \quad (5)$$

where $M(R_{train}^\beta, \theta)_{u,i}$ is the estimated rating for user u on item i for M with parameters θ being trained on the dataset R_{train}^β and $|\cdot|$ is the absolute function. Please note that the same R_{test} is utilized for different values of β .

Table 1. MetaMF’s error measurements (reproduced/original) for our five datasets alongside the MAE (mean absolute error) and the MSE (mean squared error) reported in the original paper. The non-reproducibility of the MSE on the Ciao dataset can be explained by the particularities of the MSE and the Ciao dataset. All other measurements can be reproduced (*RQ1a*).

Dataset	MAE	MSE
Douban	0.588/0.584	0.554/0.549
Hetrec-MovieLens	0.577/0.571	0.587/0.578
MovieLens 1M	0.687/0.687	0.765/0.760
Ciao	0.774/0.774	1.125/1.043
Jester	0.856/-	1.105/-

Furthermore, it is noteworthy that the magnitude of $\Delta\text{MAE}@{\beta}$ measurements does not depend on the underlying dataset, as it is a relative measure. Thus, one can compare a model’s $\Delta\text{MAE}@{\beta}$ measurements among different datasets.

2.4 Source Code and Materials

For the reproducibility study, we utilize and extend the original implementation of MetaMF, which is provided by the authors alongside the Douban, Hetrec-MovieLens, MovieLens 1M, and Ciao dataset samples via BitBucket¹. Furthermore, we publish the entire Python-based implementation of our work on GitHub² and our three user groups for all five datasets on Zenodo³ [18].

We want to highlight that we are not interested in outperforming any state-of-the-art approaches on our five datasets. Thus, we refrain from conducting any hyperparameter tuning or parameter search and utilize precisely the same parameters, hyperparameters, and optimization algorithms as Lin et al. [16].

3 Reproducibility Study

In this section, we address *RQ1a* and *RQ1b*. As such, we repeat experiments by Lin et al. [16] to verify the reproducibility of their results. Therefore, we evaluate MetaMF on the four datasets Douban, Hetrec-MovieLens, MovieLens 1M, and Ciao. Additionally, we measure its accuracy on the Jester dataset. Please note that we strictly follow the evaluation procedure as in the work to be reproduced.

We provide MAE (mean absolute error) and MSE (mean squared error) measurements on our five datasets in Table 1. It can be observed that we can reproduce the results by Lin et al. up to a margin of error smaller than 2%. Only in

¹ <https://bitbucket.org/HeavenDog/metamf/src/master/>, Last accessed Oct. 2020.

² <https://github.com/pmuellner/RobustnessOfMetaMF>.

³ <https://doi.org/10.5281/zenodo.4031011>.

the case of the MSE on the Ciao dataset, we obtain different results. Due to the selection of random batches during training, our model slightly deviates from the one utilized by Lin et al. Thereby, also, the predictions are likely to differ marginally. As described in [21], the MSE is much more sensitive to the variance of the observations than the MAE. Thus, we argue that the non-reproducibility of the MSE on the Ciao dataset can be explained by the sensitivity of the MSE on the variance of the observations in each batch. In detail, we observed in Sect. 2.2 that Ciao comprises very few ratings but lots of items. Thus, the predicted ratings are sensitive to the random selection of training data within each batch. However, it is noteworthy that we can reproduce the more stable MAE on the Ciao dataset. Hence, we conclude that our results provide strong evidence of the originally reported measurements being reproducible, enabling us to answer *RQ1a* in the affirmative.

Next, we study the rating prediction models' weights and the learned item embeddings. Again, we follow the procedure of Lin et al. and utilize the popular t-SNE (t-distributed stochastic neighborhood embedding) [17] method to reduce the dimensionality of the weights and the item embeddings to two dimensions. Since Lin et al. did not report any parameter values for t-SNE, we rely on the default parameters, i.e., we set the perplexity to 30 [17]. After the dimensionality reduction, we standardize all observations $x \in X$ by $\frac{x-\mu}{\sigma}$, where μ is the mean and σ is the standard deviation of X . The rating prediction model of each user is defined as a two-layer neural network. However, we observe that Lin et al. did not describe what layer's weights they visualize. Correspondences with the leading author of Lin et al. clarified that in their work, they only describe the weights of the first layer of the rating prediction models. The visualizations of the first layer's weights of the rating prediction models on our five datasets are given in Fig. 1.

In line with Lin et al., we discuss the weights and the item embeddings with respect to personalization and collaboration. As the authors suggest, personalization leads to distinct weight embeddings and collaboration leads to clusters within the embedding space. First, we observe that MetaMF tends to generate different weight embeddings for each user. Second, the visualizations exhibits well-defined clusters, which indicates that MetaMF can exploit collaborative information among users. However, our visualizations of the weights deviate slightly from the ones reported by Lin et al. Similar to the reproduction of the accuracy measurements in Table 1, we attribute this to the inability to derive the exact same model as Lin et al. Besides, t-SNE comprises random components and thus, generates slightly varying visualizations. However, the weights for the Ciao dataset in Fig. 1d illustrate behavior that contradicts Lin et al.'s observations. In the case of the Ciao dataset, they did not observe any form of clustering and attributed this behavior to the small number of ratings per user in the Ciao dataset. To test their claim, we also illustrate the Jester dataset with a similarly low number of ratings per user. In contrast, our visualizations indeed show well-defined clusters and different embeddings. We note that Jester exhibits many more clusters than the other datasets due to the much larger

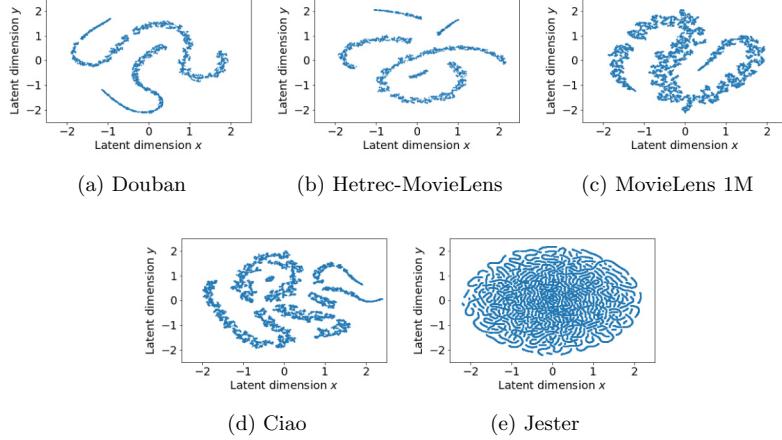


Fig. 1. MetaMF’s weights embeddings of the first layer of the rating prediction models. One observation corresponds to an individual user (*RQ1b*).

number of users. Overall, we find that both, Ciao and Jester, do not support the claim made by Lin et al. However, we see the possibility that this observation may be caused by randomness during training.

Due to space limitations, we refrain from visualizing the item embeddings. It is worth noticing that our observations on the weights also hold for the item embeddings. In detail, our visualizations exhibit indications of collaboration and personalization for all datasets. Overall, we find the visualizations of the weights and the item embeddings presented by Lin et al. to be reproducible for the Douban, Hetrec-MovieLens, and MovieLens 1M datasets and thus, we can also positively answer *RQ1b*.

4 Privacy-Focused Study

In the following, we present experiments that go beyond reproducing Lin et al.’s work [16]. Concretely, we explore the robustness of MetaMF against decreasing privacy budgets and discuss *RQ2a* and *RQ2b*. More detailed, we shed light on the effect of decreasing privacy budgets on MetaMF in two settings: (i) the role of MetaMF’s meta learning component and (ii) MetaMF’s ability to serve users with different amounts of rating data equally well.

First, we compare MetaMF to NoMetaMF in the setting of decreasing privacy budgets. Therefore, we utilize our sampling procedure in Algorithm 1 to generate datasets with different privacy budgets. In detail, we construct 10 training sets, i.e., $\{R_{train}^\beta : \beta \in \{1.0, 0.9, \dots, 0.2, 0.1\}\}$, on which MetaMF and NoMetaMF are trained on. Then, we evaluate both models on the test set R_{test} . It is worth noticing that R_{test} is the same for all values of β to enable a valid comparison. Our results in Fig. 2a illustrate that for all datasets, MetaMF preserves its

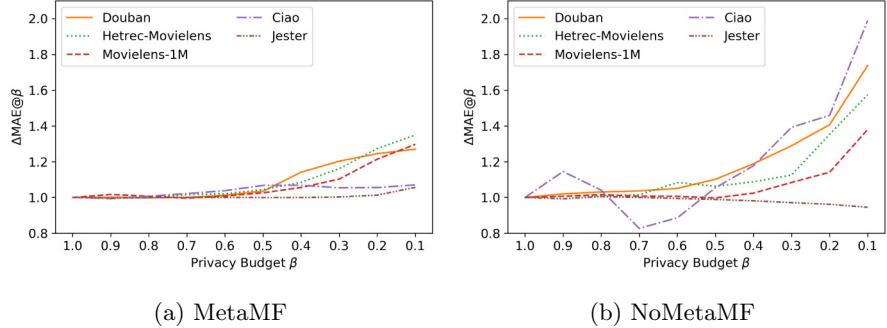


Fig. 2. $\Delta\text{MAE}@{\beta}$ measurements on (a) MetaMF and (b) NoMetaMF, in which meta learning is disabled. Especially for small privacy budgets, MetaMF yields a much more stable accuracy than NoMetaMF (*RQ2a*).

predictive capabilities well, even with decreasing privacy budgets. However, a privacy budget of $\approx 50\%$ seems to be a critical threshold. The $\Delta\text{MAE}@{\beta}$ only marginally increases for $\beta > 0.5$, but rapidly grows for $\beta \leq 0.5$ in the case of the Douban, Hetrec-MovieLens, and MovieLens 1M dataset. In other words, a user could afford to withhold $\leq 50\%$ of her data and still get well-suited recommendations. Additionally, the $\Delta\text{MAE}@{\beta}$ remains stable for the Ciao and Jester dataset. Similar observations can be made about the results of NoMetaMF in Fig. 2b. Again, the predictive capabilities remain stable for $\beta > 0.5$ in the case of Douban, Hetrec-MovieLens, and MovieLens 1M, but decrease tremendously for higher levels of privacy. Our side-by-side comparison of MetaMF and NoMetaMF in Fig. 2 suggests that both methods exhibit robust behavior for large privacy budgets (i.e., $\beta > 0.5$), but exhibit an increasing MAE for less data available (i.e., $\beta \leq 0.5$). However, we would like to highlight that the increase of the MAE is much worse for NoMetaMF than for MetaMF. Here, the $\Delta\text{MAE}@{\beta}$ indicates that the MAE for NoMetaMF increases much faster than the MAE for MetaMF for decreasing privacy budgets. This observation pinpoints the importance of meta learning and personalization in settings with a limited amount of data per user, i.e., a high privacy level. Thus, concerning *RQ2a*, we conclude that MetaMF is indeed more robust against decreasing privacy budgets than NoMetaMF, but yet, requires a sufficient amount of data per user.

Next, we compare MetaMF to NoMetaMF with respect to their ability for personalization and collaboration in the setting of decreasing privacy budgets. As explained in Sect. 3, we refer to Lin et al., which suggest that personalization leads to distinct weight embeddings and collaboration leads to clusters within the embedding space. In Fig. 3, we illustrate the weights of the first layer of the rating prediction models of MetaMF and NoMetaMF for the MovieLens 1M dataset for different privacy budgets (i.e., $\beta \in \{1.0, 0.5, 0.1\}$). Again, we applied t-SNE to reduce the dimensionality to two dimensions, followed by standardization to ease the visualization. In the case of MetaMF, we observe that it preserves the

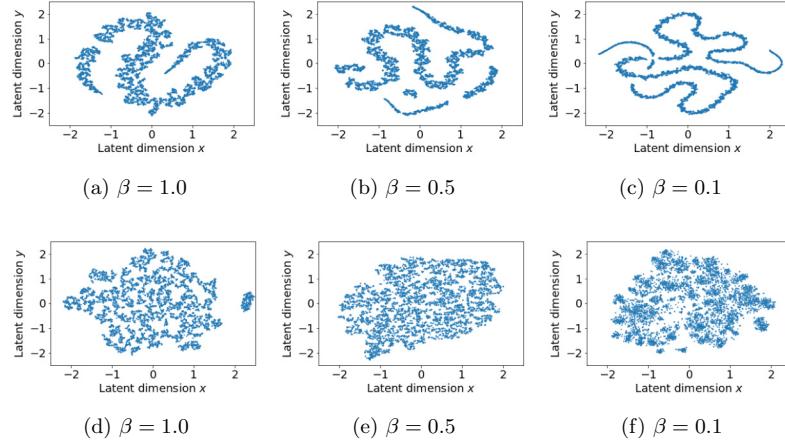


Fig. 3. Weights of the first layer of the rating prediction models for the MovieLens 1M dataset. (a), (b), (c) depict MetaMF, whereas (d), (e), (f) depict NoMetaMF, in which meta learning is disabled. No well-defined clusters are visible for NoMetaMF, which indicates the inability to exploit collaborative information among users (*RQ2a*).

ability to generate different weights for each user for decreasing privacy budgets. Similarly, well-defined clusters can be seen, which indicates that MetaMF also preserves the ability to capture collaborative information among users. In contrast, our visualizations for NoMetaMF do not show well-defined clusters. This indicates that NoMetaMF loses the ability to exploit collaborative information among users. Due to limited space, we refrain from presenting the weights of the first layer of the rating prediction models for the other datasets. However, we observe that MetaMF outperforms NoMetaMF in preserving the collaboration ability for decreasing privacy budgets on the remaining four datasets, which is also in line with our previous results regarding *RQ2a*.

In the following, we elaborate on how the high degree of personalization in MetaMF impacts the recommendations of groups of users with different amounts of rating data. In a preliminary experiment, we measure the MAE on our three user groups *Low*, *Med*, and *High* on our five datasets in Table 2. Except for the Ciao dataset, our results provide evidence that *Low* is served with significantly worse recommendations than *High*. In other words, users with lots of ratings are advantaged over users with only a few ratings.

To detail the impact of decreasing privacy budgets on these user groups, we monitor the $\Delta\text{MAE}@\beta$ on *Low*, *Med*, and *High*. The results for our five datasets are presented in Fig. 4. Surprisingly, *Low* seems to be much more robust against small privacy budgets than *High*. Here, we refer to our observations about MetaMF’s performance on the Ciao and Jester dataset in Fig. 2a. In contrast to the other datasets, Ciao and Jester comprise only a small average number of ratings per user, i.e., 38 (Ciao) and 56 (Jester), which means that they share a common property with our *Low* user group. Thus, we suspect a relationship

Table 2. MetaMF’s MAE (mean absolute error) measurements for our three user groups on the five datasets. Here, we simulated a privacy budget of $\beta = 1.0$. According to a one-tailed t-Test, *Low* is significantly disadvantaged over *High*, indicated by *, i.e., $\alpha = 0.05$ and ****, i.e., $\alpha = 0.0001$ (*RQ2b*).

Dataset	<i>Low</i>	<i>Med</i>	<i>High</i>
Douban*	0.638	0.582	0.571
Hetrec-MovieLens****	0.790	0.603	0.581
MovieLens 1M****	0.770	0.706	0.673
Ciao	0.773	0.771	0.766
Jester****	1.135	0.855	0.811

between the robustness against decreasing privacy budgets and the amount of rating data per user. The most prominent examples of *Low* being more robust than *High* can be found in Figs. 4a, 4b and 4c. Here, the accuracy of MetaMF on *High* substantially decreases for small privacy budgets. On the one hand, MetaMF provides strongly personalized recommendations for users with lots of ratings, which results in a high accuracy for these users (i.e., *High*). On the other hand, this personalization leads to a serious reliance on the data, which has a negative impact on the performance in settings with small privacy budgets. Thus, concerning *RQ2b*, we conclude that users with lots of ratings receive better recommendations than other users if they can take advantage of their abundance

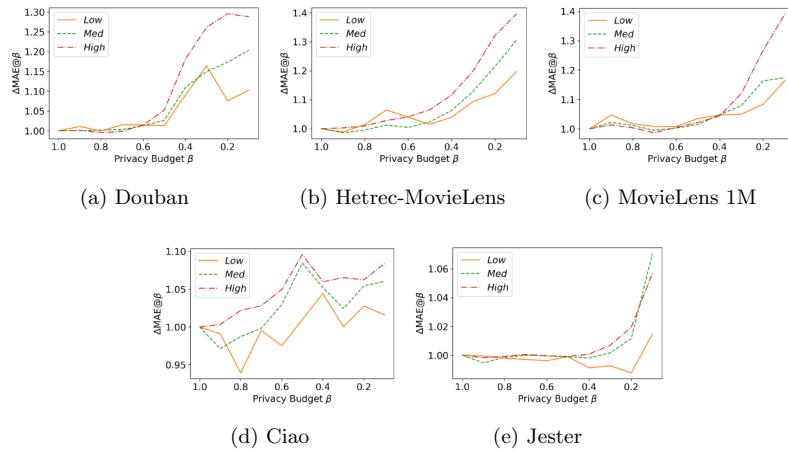


Fig. 4. MetaMF’s Δ MAE@ β measurements for the (a) Douban, (b) Hetrec-MovieLens, (c) MovieLens 1M, (d) Ciao, and (e) Jester dataset for all three usergroups. Especially (a), (b), and (c) illustrate that *High* is sensitive to small privacy budgets. In contrast, *Low* can afford a high degree of privacy, since the accuracy of its recommendations only marginally decreases (*RQ2b*).

of data. In settings where a high level of privacy is required, i.e., a low privacy budget, and thus, users decide to hold back the majority of their data, users are advantaged who do not require as much personalization from the recommender system.

5 Conclusions and Future Work

In our study at hand, we conducted two lines of research. First, we reproduced results presented by Lin et al. in [16]. Besides, we introduced a fifth dataset, i.e., Jester, which, in contrast to the originally utilized datasets, has plenty of rating data per item. We found that all accuracy measurements are indeed reproducible (*RQ1a*). However, our reproduction of the t-SNE visualizations of the embeddings illustrated potential discrepancies between our and Lin et al.'s work (*RQ1b*). Second, we conducted privacy-focused studies. Here, we thoroughly investigated the meta learning component of MetaMF. We found that meta learning takes an important role in preserving the accuracy of the recommendations for decreasing privacy budgets (*RQ2a*). Furthermore, we evaluated MetaMF's performance with respect to decreasing privacy budgets on three user groups that differ in their amounts of rating data. Surprisingly, the accuracy of the recommendations for users with lots of ratings seems far more sensitive to small privacy budgets than for users with a limited amount of data (*RQ2b*).

Future Work. In our future work, we will research how to cope with incomplete user profiles in our datasets, as users may already have limited the amount of their rating data to satisfy their privacy constraints. Furthermore, we will develop methods that identify the ratings a user should share based on the characteristics of the data.

Acknowledgements. We thank the Social Computing team for their rich feedback on this work. This work is supported by the H2020 project TRUSTS (GA: 871481) and the “DDAI” COMET Module within the COMET – Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG.

References

1. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation. In: Workshop on Recommendation in Multi-stakeholder Environments in Conjunction with RecSys 2019 (2019)
2. Ammad-Ud-Din, M., et al.: Federated collaborative filtering for privacy-preserving personalized recommendation system. arXiv preprint [arXiv:1901.09888](https://arxiv.org/abs/1901.09888) (2019)
3. Cantador, I., Brusilovsky, P., Kuflik, T.: Second international workshop on information heterogeneity and fusion in recommender systems. In: RecSys 2011 (2011)

4. Chen, C., Zhang, J., Tung, A.K., Kankanhalli, M., Chen, G.: Robust federated recommendation system. arXiv preprint [arXiv:2006.08259](https://arxiv.org/abs/2006.08259) (2020)
5. Chen, F., Luo, M., Dong, Z., Li, Z., He, X.: Federated meta-learning with fast convergence and efficient communication. arXiv preprint [arXiv:1802.07876](https://arxiv.org/abs/1802.07876) (2018)
6. Duriakova, E., et al.: PDMFRec: a decentralised matrix factorisation with tunable user-centric privacy. In: RecSys 2019 (2019)
7. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML 2017 (2017)
8. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: AIS-TATS 2011 (2011)
9. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: a constant time collaborative filtering algorithm. Inf. Retrieval **4**(2), 133–151 (2001)
10. Guo, G., Zhang, J., Thalmann, D., Yorke-Smith, N.: ETAF: an extended trust antecedents framework for trust prediction. In: ASONAM 2014 (2014)
11. Ha, D., Dai, A., Le, Q.V.: Hypernetworks. In: ICLR 2016 (2016)
12. Hahnloser, R.H., Sarpeshkar, R., Mahowald, M.A., Douglas, R.J., Seung, H.S.: Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. Nature **405**(6789), 947–951 (2000)
13. Harper, F.M., Konstan, J.A.: The movielens datasets: history and context. ACM Trans. Interact. Intell. Syst. (TIIS) **5**(4), 1–19 (2015)
14. Hu, L., Sun, A., Liu, Y.: Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In: SIGIR 2014 (2014)
15. Jiang, Y., Konečný, J., Rush, K., Kannan, S.: Improving federated learning personalization via model agnostic meta learning. In: International Workshop on Federated Learning for User Privacy and Data Confidentiality in conjunction with NeurIPS 2019 (2019)
16. Lin, Y., et al.: Meta matrix factorization for federated rating predictions. In: SIGIR 2020 (2020)
17. Maaten, L.V.D., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
18. Müllner, P., Kowald, D., Lex, E.: User Groups for Robustness of Meta Matrix Factorization Against Decreasing Privacy Budgets (2020). <https://doi.org/10.5281/zenodo.4031011>
19. Schedl, M., Bauer, C.: Distance-and rank-based music mainstreaminess measurement. In: UMAP 2017 (2017)
20. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: NIPS 2017 (2017)
21. Willmott, C.J., Matsuura, K.: Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Climate Res. **30**(1), 79–82 (2005)

P9 ReuseKNN: Neighborhood Reuse for Differentially-Private KNN-Based Recommendations (2023)

Privacy and Limited Preference Information in Recommender Systems

[P9] Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). ReuseKNN: Neighborhood Reuse for Differentially-Private KNN-Based Recommendations. *ACM Transactions on Intelligent Systems and Technology*, 14:5, pp. 1-29. DOI: <https://doi.org/10.1145/3608481>



ReuseKNN: Neighborhood Reuse for Differentially Private KNN-Based Recommendations

PETER MÜLLNER, Know-Center GmbH and Graz University of Technology, Austria

ELISABETH LEX, Graz University of Technology, Austria

MARKUS SCHEDL, Johannes Kepler University Linz and Linz Institute of Technology, Austria

DOMINIK KOWALD, Know-Center GmbH and Graz University of Technology, Austria

User-based *KNN* recommender systems (*UserKNN*) utilize the rating data of a target user's k nearest neighbors in the recommendation process. This, however, increases the privacy risk of the neighbors, since the recommendations could expose the neighbors' rating data to other users or malicious parties. To reduce this risk, existing work applies differential privacy by adding randomness to the neighbors' ratings, which unfortunately reduces the accuracy of *UserKNN*. In this work, we introduce *ReuseKNN*, a novel differentially private *KNN*-based recommender system. The main idea is to identify small but highly reusable neighborhoods so that (i) only a minimal set of users requires protection with differential privacy and (ii) most users do not need to be protected with differential privacy since they are only rarely exploited as neighbors. In our experiments on five diverse datasets, we make two key observations. Firstly, *ReuseKNN* requires significantly smaller neighborhoods and, thus, fewer neighbors need to be protected with differential privacy compared with traditional *UserKNN*. Secondly, despite the small neighborhoods, *ReuseKNN* outperforms *UserKNN* and a fully differentially private approach in terms of accuracy. Overall, *ReuseKNN* leads to significantly less privacy risk for users than in the case of *UserKNN*.

80

CCS Concepts: • Information systems → Recommender systems; Collaborative filtering; • Security and privacy → Privacy-preserving protocols;

Additional Key Words and Phrases: Neighborhood reuse, differential privacy, collaborative filtering, k nearest neighbors, recommender systems, privacy risk, popularity bias

ACM Reference format:

Peter Müllner, Elisabeth Lex, Markus Schedl, and Dominik Kowald. 2023. ReuseKNN: Neighborhood Reuse for Differentially Private KNN-Based Recommendations. *ACM Trans. Intell. Syst. Technol.* 14, 5, Article 80 (August 2023), 29 pages.

<https://doi.org/10.1145/3608481>

This research is funded by the “DDAI” COMET Module within the COMET – Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG. This research received support by the TU Graz Open Access Publishing Fund, the Austrian Science Fund (FWF): P33526 and DFH-23; and by the State of Upper Austria and the Federal Ministry of Education, Science, and Research, through grant no. LIT-2020-9-SEE-113. Authors' addresses: P. Müllner and D. Kowald, Know-Center GmbH, 8010, Sandgasse 36/4, Graz, Austria and Graz University of Technology, 8010, Rechbauerstraße 12, Graz, Austria; emails: pmuellner@know-center.at, pmuellner@student.tugraz.at, dkowald@know-center.at, dominik.kowald@tugraz.at; E. Lex, Graz University of Technology, 8010, Rechbauerstraße 12, Graz, Austria; email: elisabeth.lex@tugraz.at; M. Schedl, Johannes Kepler University Linz and Linz Institute of Technology, 4040, Altenberger Straße 69, Linz, Austria; email: markus.schedl@jku.at.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2023 Copyright held by the owner/author(s).

2157-6904/2023/08-ART80 \$15.00

<https://doi.org/10.1145/3608481>

1 INTRODUCTION

Recommender systems often rely on neighborhood-based collaborative filtering [30] to generate recommendations. These systems can intuitively justify their recommendations to the target user and also efficiently incorporate new rating data from users, which are two key issues of modern recommender systems [16]. For example, user-based KNN, i.e., *UserKNN*, is a variant of neighborhood-based collaborative filtering that utilizes the rating data of the k nearest neighbors of a target user to process a rating query. A rating query is a request to a recommender system to predict a rating for a target user to a target item. However, the way in which rating queries are processed by *UserKNN* can increase the privacy risk of users since the estimated rating scores, which determine whether an item will be recommended, are generated based on rating data of users that are used as neighbors. In this regard, existing research [9, 49, 64] finds that these neighbors are susceptible to multiple privacy risks, such as the inference of their private rating data (see Section 3). To mitigate that privacy risk, several works [10, 24, 65] use *differential privacy (DP)* [18, 20] to protect users' rating data by adding a degree of randomness to the data. However, the added randomness typically leads to severe drops in recommendation accuracy [7].

To address this problem, we introduce *ReuseKNN*, a novel differentially private KNN-based recommender system that reduces the number of neighbors to which differential privacy needs to be applied. Intuitively, instead of utilizing new users as neighbors for processing new rating queries, *ReuseKNN* reuses useful neighbors from past rating queries. Hence, *ReuseKNN* constructs small but highly reusable neighborhoods for every target user by fostering the neighbors' reusability for many rating queries. With this, as illustrated in Figure 1, *ReuseKNN* minimizes the set of users that need to be protected with DP—we call them “vulnerable users”. Plus, most users do not need to be protected with DP, as their rating data is only rarely used in the recommendation process—we call them “secure users”. As shown, we also introduce a data usage threshold τ , i.e., a hyperparameter that allows adjusting the maximum data usage for a user to be treated as secure. In this way, we leave it to the recommender system provider to specify what degree of data usage is tolerated despite the resulting privacy risks and which users need to be protected.

We evaluate the proposed approach in a two-stage procedure: (i) neighborhood reuse only, i.e., *ReuseKNN*, and (ii) neighborhood reuse with DP, i.e., *ReuseKNN_{DP}*. In the first stage, *ReuseKNN* does not use DP at all. With this, we focus on how neighborhood reuse can increase the reusability of neighbors and preserve *UserKNN*'s recommendation accuracy. In the second stage, we combine *ReuseKNN* with DP, i.e., *ReuseKNN_{DP}*, to protect vulnerable users with DP. This allows the investigation of how *ReuseKNN_{DP}* can mitigate all users' privacy risk while generating accurate recommendations. We evaluate *ReuseKNN* and *ReuseKNN_{DP}* on five different datasets: *MovieLens 1M*, *Douban*, *LastFM*, *Ciao*, and *Goodreads*. Plus, we compare *ReuseKNN* and *ReuseKNN_{DP}* with five KNN-based baselines that utilize DP (e.g., [65]) and the concept of neighborhood reuse in different ways with respect to recommendation accuracy and users' privacy risk. Additionally, the nature of neighborhood reuse may raise concerns that the generated recommendations are biased towards items consumed by many users, i.e., popular items. Thus, we investigate whether the proposed approach is more or less prone to item popularity bias than the baselines.

Our results indicate that *ReuseKNN* yields significantly smaller neighborhoods than traditional *UserKNN*. Despite the smaller neighborhoods, *ReuseKNN* and *ReuseKNN_{DP}* outperform our baselines in terms of recommendation accuracy. Moreover, *ReuseKNN_{DP}* leads to significantly less privacy risk for users than *UserKNN* with DP. Also, the proposed approach does not increase item popularity bias. Overall, the three main contributions of this article are as follows:

- (1) We present a novel *ReuseKNN* recommender system and compare two neighborhood reuse strategies to substantially foster the reusability of a target user's neighborhood and effectively reduce the number of vulnerable users.

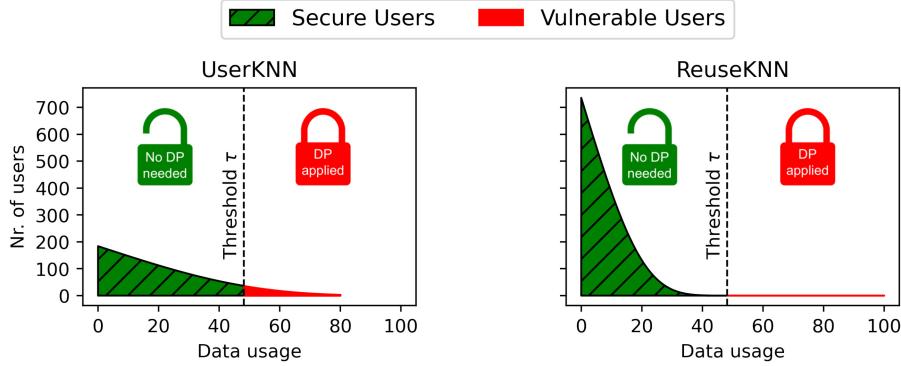


Fig. 1. Schematic illustration of the data usage (i.e., how often a user is used as a neighbor) distribution of traditional *UserKNN* and the proposed *ReuseKNN* recommender system. *ReuseKNN* increases the number of secure users (green, no differential privacy needed) and decreases the number of vulnerable users (red, differential privacy needs to be applied) compared with *UserKNN*. The dashed line illustrates the data usage threshold τ , a hyperparameter for adjusting the maximum data usage for a user to be treated as secure.

- (2) We combine *ReuseKNN* with DP to realize *ReuseKNN_{DP}* and show that *ReuseKNN_{DP}* improves recommendation accuracy over KNN- and DP-based baselines and, at the same time, does not increase item popularity bias.
- (3) We show that *ReuseKNN_{DP}* leads to significantly less privacy risk, since most users are rarely exploited in the recommendation process and only the remaining users, i.e., vulnerable users, are protected with DP.

Our work illustrates how to address privacy risks in KNN-based recommender systems through neighborhood reuse combined with DP. While the proposed approach focuses on traditional KNN, we additionally demonstrate the generalizability of the neighborhood reuse principle to user and item embeddings created by state-of-the-art neural collaborative filtering approaches [29].

2 RELATED WORK

We describe two research strands related to our work: (i) studies on the identification and quantification of users' privacy risks in recommender systems and (ii) privacy-aware recommender systems that mitigate users' privacy risks. Since *ReuseKNN* is a differentially private and KNN-based recommender system, we emphasize KNN-based methods when reviewing privacy risks in recommender systems as well as DP when reviewing privacy-preserving technologies for recommender systems. Also, we focus on the privacy risks that arise from the recommendations presented to potentially malicious target users. This can harm the neighbors used in the recommendation process.

2.1 Privacy Risks in Recommender Systems

Previous research [5, 23, 36, 49] describes many severe privacy risks for users of recommender systems. For example, according to Ramakrishnan et al. [49], the use of neighbors' rating data in the recommendation process can pose a privacy risk to the neighbors. Serendipitous recommendations could reveal unique connections between neighbors and items. In this way, the rating data of the neighbors can be uncovered or the neighbors' identities can be revealed within the recommendation database. Also, Zhang et al. [64] show that it could be possible to identify users whose data was used in the recommendation process. Their results suggest that the effectiveness of their attack depends on the number of generated recommendations. Moreover,

Calandrino et al. [9] propose to generate fake users, i.e., sybils, based on limited knowledge of a victim's data. These sybils can isolate the victim that is utilized as a neighbor and compromise its privacy.

To quantify users' privacy risks in computational systems such as recommender systems, several privacy risk metrics [13, 17, 42, 53, 56] have been proposed. These metrics often rely on the sensitivity of users' data, i.e., how strong this data puts users' privacy at risk. For example, Chen et al. [13] detect correlations within the dataset to measure whether a piece of data could reveal personal information about the users. Srivastava and Geethakumari [53] measure the relative sensitivity of a single piece of data compared with the remaining data of a user. Similarly, Domingo-Ferrer [17] relates the overall sensitivity of a user's data to the sensitivity of other users' data. Liu and Terzi's *privacy score* [42] weighs the sensitivity with the degree of visibility of a user's data (i.e., how often a user's data is utilized in the recommendation process).

Evaluating the privacy risk of users based on attacks only measures the privacy risk with respect to the specific attack scenario. Liu and Terzi's metric measures users' privacy risk independent of specific attack scenarios and, thus, allows investigating privacy risk in a recommender system at a more general level. Therefore, in our work, we utilize Liu and Terzi's metric to measure users' privacy risk in a general neighborhood-based recommendation scenario. Furthermore, we assume that all pieces of data are equally sensitive, since sensitivity typically depends on the application and the user's perception of privacy [38].

2.2 Privacy-Aware Recommender Systems

Several works [33, 55, 63] mitigate users' privacy risks by applying *homomorphic encryption* [25] to users' rating data. Here, recommendations are generated based on the encrypted rating data, and, thus, users' rating data remains protected in the recommendation process. Homomorphic encryption, however, has high computational complexity. Thus, Tang and Wang [55] apply homomorphic encryption on the rating data of a target users' friends only, i.e., a small subset of users, to improve computational efficiency. Besides homomorphic encryption, *federated learning* [44] is used to lower users' privacy risks [27, 41, 48, 60]. Specifically, instead of a user's rating data, the parameters of the user's local recommendation model are utilized in the recommendation process. For example, Perifanis and Efraimidis [48] combine federated learning with neural collaborative filtering [29] to improve privacy. However, since federated learning could still leak user data [47, 50], research proposes to learn a user's local model by utilizing only a subset of the rating data [4, 14, 46]. Moreover, *differential privacy (DP)* [18, 20] has been leveraged for collaborative filtering recommender systems [10–12, 24, 60, 65]. These techniques add randomness to users' data to hide the actual data. Therefore, they face a trade-off between accuracy and privacy (e.g., [7]). To address this trade-off, Xin and Jaakkola [61] assume a moderate number of public users who tolerate disclosing their rating data. With this unprotected rating data, recommendation accuracy can be preserved while the privacy requirements of the remaining users are respected.

It has been shown in several studies [1, 39, 43] that users often receive more recommendations for popular items, and correspondingly non-popular items receive less exposure. This behavior of recommender systems, which is referred to as *popularity bias*, leads to disparate, i.e., unfair, treatment of less popular items. Dwork et al. [19] and Zemel et al. [62] show that, formally, there is a close connection between fairness and DP. However, the sole application of DP is insufficient to ensure fairness due to correlations within the dataset [21]. Moreover, Ekstrand et al. [21] and Agarwal [3] highlight a trade-off between user privacy and fairness. Overall, related work suggests that DP can severely impact recommendations in different ways, for example, result in popularity

Table 1. Overview of the Notation Used in this Article

Symbol	Description
k	Number of neighbors to process a rating query for target user u and target item i .
Q_u	Rating queries for target user u , i.e., the items in u 's test set R_u^{test} .
\mathcal{R}^k	User-based KNN recommender system utilizing k neighbors to predict ratings.
$\mathcal{R}^k(u, i)$	Estimated rating score for target user u and target item i by recommender system \mathcal{R}^k .
$\mathcal{R}_{top}^k(u)$	Items with the highest estimated rating score for target user u .
$r_{u,i}$	Rating score of user u to item i .
U	The set of users.
U_i	The set of users that rated item i .
I	The set of items.
I_u	The set of items rated by user u .
R	The set of ratings.
$N_{u,i}^k$	The k nearest neighbors for target user u and target item i .
$N_{u,i}$	Neighbors of target user u and rated item i .
N_u	The set of neighbors for target user u across all rating queries.
$N_u^{(q)}$	The set of neighbors for target user u across q rating queries.
$sim(u, n)$	Similarity score between target user u and neighbor n .
$reusability(c u)$	Reusability score of candidate neighbor c for target user u .
$ranking(\cdot)$	The ranking function that ranks candidate neighbors w.r.t. similarity and reusability.
τ	Data usage threshold, i.e., the maximal usage of a user's data that is tolerated.
m_{DP}	Differential privacy mechanism that utilizes plausible deniability.
ϵ	Privacy parameter.
S	Secure users that do not need to be protected with DP.
V	Vulnerable users that need to be protected with DP.
R_S	Rating data of secure users.
\tilde{R}_V	DP-protected rating data of vulnerable users.
α	Significance level used for the statistical tests.
σ_x	Sample standard deviation of variable x .
$\sigma_{x,y}$	Sample covariance of variables x and y .

bias. Therefore, we believe that it is important to evaluate the proposed approach, *ReuseKNN*, also in terms of item popularity bias.

Similar to our work, previous research by Zhu et al. [65] prevents the inference of neighbors' rating data by applying DP to the users' rating data in *UserKNN*. However, to preserve recommendation accuracy, Zhu et al. vary the degree of randomness that is added to all users' rating data based on the sensitivity of the data. In contrast, *ReuseKNN* preserves recommendation accuracy by adding randomness only where it is necessary, i.e., to vulnerable users with a high privacy risk. In the remainder of the article, we use a variant of the approach of Zhu et al. that is comparable to the proposed approach as baseline (i.e., $UserKNN_{DP}^{full}$) for our experiments.

3 PROBLEM DEFINITION

In the following, we discuss one key vulnerability of *UserKNN*, which poses privacy risks to the neighbors utilized in the recommendation process. Also, we precisely model the adversary's goal, i.e., the inference of the neighbors' rating data. A summary of the notation used in this article is given in Table 1.

3.1 Vulnerability Analysis of UserKNN

Typically, a user-based KNN recommender system \mathcal{R}^k , i.e., *UserKNN*, generates an estimated rating score for a rating query of a target user u and a target item i by utilizing the ratings of k other

users that have rated i , i.e., the k nearest neighbors $N_{u,i}^k$:

$$\mathcal{R}^k(u, i) = \frac{\sum_{n \in N_{u,i}^k} sim(u, n) \cdot r_{n,i}}{\sum_{n \in N_{u,i}^k} sim(u, n)}, \quad (1)$$

where $sim(u, n)$ is the similarity between target user u and neighbor n , commonly determined via Pearson's correlation coefficient [6] or Cosine similarity between the users' rating vectors. For *UserKNN*, the neighborhood $N_{u,i}^k$ used for generating recommendations for target user u and item i , comprises the k most similar neighbors:

$$N_{u,i}^k = \arg \max_{c \in U_i}^k sim(u, c), \quad (2)$$

where U_i are all users that have rated i and sim is the similarity metric. *UserKNN* utilizes the rating data of the target user's k nearest neighbors to generate an estimated rating score (see Equation (1)). Therefore, the estimated rating score $\mathcal{R}^k(u, i)$ for target user u and item i is linked to the neighbors' rating data. Through learning the behavior of *UserKNN*, the estimated rating score could reveal the rating data of users that have been used as neighbors [9]. Therefore, the privacy threat for users can be traced back to them being utilized as neighbors in the recommendation process.

3.2 Attack Model

In this work, we assume that a user with malicious intent, i.e., the *adversary* a , exploits the vulnerability above via querying estimated rating scores from the recommender system, i.e., $\mathcal{R}^k(a) = \{\mathcal{R}^k(a, i_1), \mathcal{R}^k(a, i_2), \dots, \mathcal{R}^k(a, i_l)\}$, where $\mathcal{R}^k(a, i_j)$ is the estimated rating score for item $i_j \in Q_a$ and Q_a is the set of a 's queries. The adversary a can target a specific user n by increasing the likelihood of n being used as neighbor. To achieve this, a would modify its own user profile R_a such that it (partially) matches n 's profile. Moreover, a can exploit publicly available data P , e.g., public rating data, product reviews, tweets, or lists of similar items, to better learn the behavior of *UserKNN* [9]. Given these assumptions, the adversary aims to infer the rating data of a neighbor n used to generate the estimated rating scores:

$$Pr[r_{n,i_1}, r_{n,i_2}, \dots, r_{n,i_l} | \mathcal{R}^k(a, i_1), \mathcal{R}^k(a, i_2), \dots, \mathcal{R}^k(a, i_l), P \cup R_a], \quad (3)$$

where r_{n,i_j} is the rating score of neighbor n for item i_j . Note that if a user is used as neighbor for many rating queries, many ratings could be targeted by an adversary. Thus, the degree to which a user's rating data is used in the recommendation process is an important indicator of this user's privacy risk (see the DataUsage@ k metric in Section 5.2.3).

Given this attack model, the privacy threat lies on the rating level, i.e., the inference of neighbors' rating scores. Therefore, our approach aims at protecting the neighbors' rating scores. In the remainder of this work, we evaluate our approach in a rating-prediction task, since this fits well to our problem statement above (see Appendix B for results of a ranking-based experiment).

4 APPROACH

In the following, we first schematically illustrate *UserKNN*'s and *ReuseKNN*'s recommendation process based on an illustrative example. Then, we outline the two neighborhood reuse strategies of the *ReuseKNN* recommender system (Section 4.2). Finally, we present *ReuseKNN_{DP}*, i.e., neighborhood reuse with *differential privacy (DP)* (Section 4.3).

4.1 Example of the Recommendation Process in *UserKNN* and *ReuseKNN*

Figure 2 provides a schematic illustration of *UserKNN*'s and *ReuseKNN*'s recommendation process, showing the interplay between a user's data usage and the user's privacy risk. For simplicity, we



Fig. 2. Schematic illustration of the recommendation process for three rating queries in Alice's query set Q_{Alice} for *UserKNN* and *ReuseKNN*. A green shaded item indicates that the rating score for this item is estimated for the target user and a red shaded item indicates that the rating score of a neighbor has been utilized for the rating estimation. Traditional *UserKNN* selects those users as neighbors that rated the queried item and have the highest similarity value; in this toy example, those are Bob and Amy. Thus, Bob and Amy are vulnerable and need to be protected with DP. In contrast, *ReuseKNN* utilizes Tim as neighbor. As such, *ReuseKNN* substantially increases reusability (5.15 instead of 1.2 and 0.74) at the price of a slightly reduced similarity (0.90 instead of 0.98 and 0.97). This way, only Tim is vulnerable and is the only neighbor that needs to be protected with DP, as Bob and Amy remain unused.

assume that Bob, Amy, and Tim have been used as neighbors for τ rating queries, i.e., data usage and privacy risk is τ . To process Alice's rating queries for items i_l and i_m , *UserKNN* selects Bob and Amy as neighbors, as they have the highest similarity values across all users that rated the queried items. Due to the usage of Bob's and Amy's data, their data usage exceeds threshold τ and DP needs to be applied. For the rating query for item i_n , again, Amy is utilized in the recommendation process. Since she is already protected with DP, her privacy risk remains at τ . This is different from how *ReuseKNN* processes rating queries. For the rating queries for items i_l , i_m , and i_n , *ReuseKNN* selects Tim as neighbor, as Tim has a substantially higher reusability value and only marginally smaller similarity than Bob and Amy. Therefore, only Tim's data usage exceeds τ , and DP is needed to protect Tim.

In summary, in this illustrative example, *UserKNN* leads to two vulnerable users, Bob and Amy, that need to be protected with DP. In contrast, *ReuseKNN* leads to only one vulnerable user, Tim, to which DP has to be applied.

4.2 ReuseKNN

The key feature of *ReuseKNN* is to reuse neighbors from a target user u 's previous rating queries to minimize the cardinality of the neighborhood $N_u = \bigcup_{i \in Q_u} N_{u,i}^k$ across all rating queries Q_u . As illustrated in Figure 1, this means that *ReuseKNN* decreases the data usage for most users, i.e., secure users, and in this way, also their privacy risk. Plus, *ReuseKNN* decreases the number of highly reused neighbors, i.e., vulnerable users with high data utilization and, thus, high privacy risk.

In addition to the similarity, *ReuseKNN* also considers the extent to which a target user u could reuse candidate neighbor c as a neighbor for many rating queries, i.e., $\text{reusability}(c|u)$. Since both similarity and reusability scores are differently distributed across their respective numeric ranges, we rank candidate neighbors according to their scores. Formally, for a user u , the rank $\text{ranking}(u) = |\{v \in U \setminus \{u\} : f(v) \leq f(u)\}|$, where U is the set of all users and f measures the *similarity* or *reusability* score. Note that $\text{ranking}(u) > \text{ranking}(v)$ if $f(u) > f(v)$ for users u and v , and that $\text{ranking}(u) = \text{ranking}(v)$ in case $f(u) = f(v)$. With this, the k neighbors $N_{u,i}^k$ are selected based on similarity and reusability. Formally:

$$N_{u,i}^k = \arg \max_{c \in U_i} [\text{ranking}(\text{sim}(u, c)) + \text{ranking}(\text{reusability}(c|u))], \quad (4)$$

where U_i are all users that rated item i , sim measures the similarity between two users, and reusability depends on the given neighborhood reuse strategy of *ReuseKNN*. In the case in which multiple candidate neighbors have equal values for $\text{ranking}(\text{sim}(u, c)) + \text{ranking}(\text{reusability}(c|u))$, we choose these neighbors at random.

To estimate a candidate neighbor's *reusability* score, *ReuseKNN* utilizes two neighborhood reuse strategies: *Expect* and *Gain*. The unpersonalized *Expect* strategy measures a candidate neighbor's reusability for an average target user, whereas the personalized *Gain* strategy measures the reusability for a specific target user. Next, we discuss two strategies to increase the reusability of a target user's neighbors: unpersonalized and personalized neighborhood reuse.

Unpersonalized Neighborhood Reuse: Expect. The more users rated an item, the more likely it is that a random target user will query a rating estimation for this item. Following this intuition, *Expect* promotes candidate neighbors that rated many popular items and penalizes candidate neighbors that either rated only a few items or many unpopular items. For *Expect*, the reusability score of candidate neighbor c is defined by

$$\text{reusability}(c|u) = \text{reusability}(c) = \sum_{i \in I_c} \frac{|U_i|}{|U|}, \quad (5)$$

where u is the target user, I_c are the items c rated, U_i are the users that rated an item i , and U is the set of all users. In this case, $\text{reusability}(c)$ is the summed-up popularity of c 's rated items and measures the *expected* number of a random user's rating queries for which c could be used as a neighbor. This means that the reusability of a candidate neighbor is estimated for an average user and not for a specific target user (i.e., unpersonalized).

Personalized Neighborhood Reuse: Gain. In contrast to unpersonalized neighborhood reuse, *Gain* measures a candidate neighbor's reusability for a specific target user. Specifically, *Gain* quantifies how many of a target user's ratings a candidate neighbor could have covered in the past, i.e., how many ratings the target user could have *gained* from the candidate neighbor. Thus, *Gain* gives the fraction of a target user u 's rated items for which a candidate neighbor c could have served as a neighbor:

$$\text{reusability}(c|u) = \frac{|I_u \cap I_c|}{|I_u|}, \quad (6)$$

where I_u are the items rated by u and I_c are the items rated by c . In contrast to the unpersonalized *Expect* strategy, *Gain*'s reusability score depends on a specific target user (i.e., personalized).

4.3 ReuseKNN_{DP}: Neighborhood Reuse and Differential Privacy

ReuseKNN leads to a minimal number of highly reused neighbors, i.e., vulnerable users, who are utilized more often as neighbors than the data usage threshold τ would allow. *ReuseKNN_{DP}* addresses this high privacy risk resulting from the frequent usage of vulnerable users (see Section 3) by adding DP to our neighborhood reuse strategies. Specifically, for a rating query for user u and item i , a privacy mechanism m_{DP} is applied to the ratings for i of vulnerable users V that are used as neighbors, i.e., $\tilde{R}_V = \{m_{DP}(r_{n,i}) : n \in N_{u,i}^k \cap V\}$. In this way, *ReuseKNN_{DP}* utilizes real ratings of secure users S , i.e., $R_S = \{r_{n,i} : n \in N_{u,i}^k \cap S\}$, plus the modified ratings \tilde{R}_V of vulnerable users, to generate the estimated rating score $\mathcal{R}^k(u, i)$:

$$\mathcal{R}^k(u, i) = \frac{\sum_{n \in N_{u,i}^k \cap S} sim(u, n) \cdot r_{n,i} + \sum_{n \in N_{u,i}^k \cap V} sim(u, n) \cdot m_{DP}(r_{n,i})}{\sum_{n \in N_{u,i}^k} sim(u, n)}. \quad (7)$$

Specifically, the privacy mechanism m_{DP} utilizes *randomized responses* [59] to achieve DP [20]. With this, intuitively, neighbors can plausibly deny that their real rating was used in the recommendation process. The privacy mechanism m_{DP} flips a fair coin and if the coin is heads, the vulnerable neighbor's real rating is utilized in the recommendation process. If the coin is tails, m_{DP} flips a second fair coin to decide whether to utilize the vulnerable neighbor's real rating or a random rating drawn from a uniform distribution over the range of ratings. With this, the adversary is unaware whether the utilized rating is real, or random, which leads to the privacy guarantees within the DP framework [20]:

$$\frac{Pr[\text{Adversary's assumption: Real rating} | \text{Truth: Real rating}]}{Pr[\text{Adversary's assumption: Real rating} | \text{Truth: Random rating}]} = \frac{0.75}{0.25} = 3 \leq e^\epsilon, \quad (8)$$

which results in a privacy parameter of $\epsilon = \ln 3$. Reconsidering user-based KNN's vulnerability (see Equation (1)), this means that if a neighbor n is considered as vulnerable, the DP-protected rating is used in the recommendation process instead of the real rating for item i (see Equation (7)). This impacts the adversary a 's objective (see Equation (3)) of inferring n 's rating data given the estimated rating scores for which n was used as neighbor and its own rating data R_a in combination with public knowledge P (see Section 3). Since a maximum of τ (i.e., the data usage threshold) real ratings of n are used by the recommender system, the remaining ratings are DP-protected. Thus, the adversary is not aware of whether the inferred rating data is the original rating data or random rating data as generated by the m_{DP} mechanism:

$$Pr[r_{n,i_1}, \dots, r_{n,i_\tau}, m_{DP}(r_{n,i_{\tau+1}}), \dots, m_{DP}(r_{n,i_L}) | \mathcal{R}^k(a, i_1), \mathcal{R}^k(a, i_2), \dots, \mathcal{R}^k(a, i_L), P \cup R_a], \quad (9)$$

where r_{n,i_j} is n 's rating for item i_j and $\mathcal{R}^k(a, i_j)$ is the estimated rating score of i_j for adversary a . Through combining non-DP and DP ratings, *ReuseKNN_{DP}* yields the following privacy parameter ϵ for each of a vulnerable user's, in this case n , utilized ratings (for details, see Appendix A):

$$\epsilon = \ln \left(3 + 4 \cdot \frac{Pr[\text{Non-DP rating}]}{Pr[\text{DP rating}]} \right). \quad (10)$$

In this way, *ReuseKNN_{DP}* combines neighborhood reuse with DP to reduce the number of neighbors to which DP needs to be applied and to ensure privacy. Overall, *ReuseKNN_{DP}* can use two neighborhood reuse strategies with DP (for details, see Section 4.2):

- (1) *Expect_{DP}*: Unpersonalized neighborhood reuse combined with DP
- (2) *Gain_{DP}*: Personalized neighborhood reuse combined with DP

5 EXPERIMENTAL SETUP

We utilize a two-stage evaluation procedure to compare and evaluate the two neighborhood reuse strategies of (i) *ReuseKNN* and (ii) *ReuseKNN_{DP}*:

Neighborhood Reuse without DP: ReuseKNN. In the first stage, we evaluate *ReuseKNN* without protecting vulnerable neighbors with DP in order to better understand the advantages and disadvantages of the proposed neighborhood reuse strategies. Hence, we compare *Expect* and *Gain* to distill the impact of neighborhood reuse for recommendations.

Neighborhood Reuse with DP: ReuseKNN_{DP}. In the second stage, we combine *ReuseKNN* with DP to protect vulnerable users, i.e., *ReuseKNN_{DP}*. We compare our neighborhood reuse strategies *Expect_{DP}* and *Gain_{DP}* to investigate how *ReuseKNN_{DP}* can address the accuracy–privacy trade-off.

5.1 Baselines

We compare *ReuseKNN* and *ReuseKNN_{DP}* with five different KNN-based baselines. Concretely, for *ReuseKNN*, i.e., neighborhood reuse without DP, we use two non-DP baselines:

- (1) *UserKNN* [30]: Traditional *UserKNN* without neighborhood reuse. No users are protected with DP (Vulnerable users $V = \emptyset$).
- (2) *UserKNN+Reuse*: A variant of *UserKNN* with neighborhood reuse. Initially, for the first rating query, e.g., for item j , the k most similar users that rated j are selected as neighbors, as in case of *UserKNN*. However, for the following rating queries, e.g., for item i and user u , $k^{prev} = \min\{k, |N_{u,i}|\}$ neighbors from all previous rating queries that rated i (i.e., $N_{u,i}$) are reused. If too few previous neighbors rated i , i.e., $k^{prev} < k$, a minimal set of $k^{new} = k - k^{prev}$ new neighbors is additionally used, as given by:

$$N_{u,i}^k = \arg \max_{n \in N_{u,i}} sim(u, c) \cup \arg \max_{c \in U_i \setminus N_{u,i}} sim(u, c), \quad (11)$$

where U_i are all users that rated item i . Similar to *UserKNN*, *UserKNN+Reuse* assumes that no users are vulnerable ($V = \emptyset$). Thus, no users are protected with DP.

For *ReuseKNN_{DP}*, i.e., neighborhood reuse with DP, we use three DP baselines:

- (1) *UserKNN_{DP}*: A variant of *UserKNN*, but DP is applied to vulnerable users $V = \{u \in U : DataUsage@k(u) > \tau\}$. See Section 5.5 for the exact τ values.
- (2) *UserKNN+Reuse_{DP}*: A variant of *UserKNN+Reuse*, but DP is applied to vulnerable users $V = \{u \in U : DataUsage@k(u) > \tau\}$. See Section 5.5 for the exact τ values.
- (3) *UserKNN_{DP}^{full}*: Traditional differentially private *UserKNN*, where DP is applied to the full set of users, i.e., $V = \{u \in U : DataUsage@k(u) \geq 0\}$ (similar to the rating perturbation in [65]).

To evaluate *ReuseKNN_{DP}*, we use the three DP baselines, as well as non-DP *UserKNN*. With this, we can compare *ReuseKNN_{DP}* to two contrastive baselines: *UserKNN_{DP}^{full}*, which protects all users with DP, and *UserKNN*, which does not apply DP at all.

5.2 Evaluation Metrics

We test the proposed approach in two evaluation stages using the following evaluation criteria and metrics (see Table 2 for an overview):

5.2.1 *Neighborhood Reuse*. To evaluate the degree to which *ReuseKNN* can reuse neighbors from previous rating queries, we measure the size of a target user’s neighborhood after multiple queries. Plus, we study whether the reused neighborhoods are capable of generating meaningful

Table 2. Overview of the Seven Evaluation Metrics Used in this Work

Evaluation Criterion	Evaluation Metric	Objective	Short Description	Evaluation Stage
				<i>ReuseKNN</i> <i>ReuseKNN_{DP}</i>
Neighborhood Reuse	Neighbors@ q	↘	Neighborhood size	•
	CoRatings@ q	↗	No. of co-rated items	•
Accuracy	MAE@ k	↘	Mean absolute error	• •
	V	↘	Percentage of vulnerable users	•
Privacy	PrivacyRisk@ k	↘	Privacy risk of users	•
	PP-Corr@ k	↘	Positivity–popularity correlation	•
Popularity Bias	Coverage@ k	↗	Percentage of item coverage	•

↘ indicates that lower values are better and ↗ indicates that higher values are better. q is the number of queries and k is the number of neighbors. With •, we indicate the evaluation stage in which the metric is used.

recommendations via measuring the number of co-rated items between the neighborhood and the target user.

Neighborhood Size. For every rating query of a target user u , k neighbors are required to generate the recommendation. In the worst case, no neighbors from previous rating queries can be reused. Thus, after q queries, $|N_u| = \min\{q \cdot k, |U| - 1\}$ for U being the set of all users. In the best case, u reuses the same k neighbors for all q queries, i.e., $|N_u| = k$. To quantify how many of u 's neighbors are reused, we measure the size of u 's neighborhood after q rating queries:

$$\text{Neighbors}@q(u) = |N_u^{(q)}|, \quad (12)$$

where $N_u^{(q)}$ is u 's set of neighbors after q rating queries. With that, we test how well our neighborhood reuse strategies of *ReuseKNN*, i.e., neighborhood reuse only, can reuse a target user's neighbors for multiple rating queries.

Number of Co-Ratings. The utilization of fewer neighbors across many rating queries might impact the accuracy of recommendations. Therefore, we test whether a target user's neighbors are beneficial for recommendation accuracy, i.e., “reliable”. One important characteristic of these reliable neighbors is the number of co-rated items with the target user [2, 16]. Thus, we measure the average number of co-rated items between a target user u and its neighbors $n \in N_u$ after q rating queries:

$$\text{CoRatings}@q(u) = \frac{1}{|N_u^{(q)}|} \sum_{n \in N_u^{(q)}} |I_u \cap I_n|, \quad (13)$$

where I_u are the items rated by target user u and I_n are the items rated by neighbor n . With this, we test how beneficial the neighborhoods are for generating accurate recommendations.

5.2.2 Accuracy. To quantify the accuracy of a target user's recommendations, we rely on the widely used mean absolute error metric (MAE). We use MAE to measure how accurate the rating scores can be predicted, because of the way in which we apply DP, i.e., via adding noise to the neighbors' rating values in order to protect against the disclosure of these ratings (see Section 3). According to Herlocker et al. [30], the number of neighbors k has an impact on the recommendation accuracy. Thus, we test the accuracy of u 's recommendations for $k \in \{5, 10, 15, 20, 25, 30\}$. Therefore, MAE@ $k(u)$ quantifies the accuracy of u 's recommendations when k neighbors are used to generate a recommendation. More formally:

$$\text{MAE}@k(u) = \frac{1}{|R_u^{test}|} \sum_{r_{u,i} \in R_u^{test}} |r_{u,i} - \mathcal{R}^k(u, i)|, \quad (14)$$

where the predicted rating score $\mathcal{R}^k(u, i)$ is compared with the real rating scores $r_{u,i} \in R_u^{test}$ in u 's test set. We note that the items for which R_u^{test} comprises ratings are the ones that are in u 's set of rating queries Q_u . We use the MAE@ $k(u)$ metric for evaluating both, *ReuseKNN*, i.e., neighborhood reuse only, and *ReuseKNN_{DP}*, i.e., neighborhood reuse with DP.

5.2.3 Privacy. Liu and Terzi [42] provide a framework to measure a user's privacy risk in computational systems, such as recommender systems based on the visibility of the user's data. In our work, we relate this visibility to how often a user's rating data was utilized in the recommendation process. As such, the DataUsage@ $k(u)$ metric counts for how many rating queries a user u was used as a neighbor. Similar to MAE@ $k(u)$, we also relate the usage of u 's data to the number of neighbors k used to generate recommendations. Formally:

$$\text{DataUsage}@k(u) = \sum_{v \in U} \sum_{i \in Q_v} \mathbb{1}_{N_{v,i}^k}(u), \quad (15)$$

where U is the set of all users, Q_v is the set of items for which user v queries estimated ratings, and $\mathbb{1}_{N_{v,i}}(u)$ is the indicator function of user u being in v 's set of neighbors $N_{v,i}$ for an item i .

Percentage of Vulnerable Users. As mentioned earlier, the main goal of neighborhood reuse is to reduce the number of users that need to be protected with DP. The DataUsage@ k definition allows us to identify these vulnerable users V , i.e., the set of users whose data is utilized more often than the adjustable privacy risk threshold τ allows:

$$V = \{u \in U : \text{DataUsage}@k(u) > \tau\}, \quad (16)$$

where U is the set of all users. Thus, the percentage of vulnerable users relates to what fraction of users DP has to be applied to (i.e., $|V|/|U|$). We use this metric to evaluate *ReuseKNN*, i.e., neighborhood reuse only.

Privacy Risk. We apply DP to a user u 's data if $\text{DataUsage}@k(u) > \tau$. This way, only the first τ utilized ratings contribute to u 's privacy risk, since for the remaining ratings that are utilized, privacy is guaranteed via the DP framework (see Section 4.3):

$$\text{PrivacyRisk}@k(u) = \min[\tau, \text{DataUsage}@k(u)]. \quad (17)$$

We use PrivacyRisk@ k to measure the users' privacy risk when neighborhood reuse is combined with DP, i.e., *ReuseKNN_{DP}*.

5.2.4 Item Popularity Bias. One might be concerned that neighborhood reuse could lead to exploiting users as neighbors that rated many popular items, which could result in more positive estimated rating scores for popular items. To test for this item popularity bias, we analyze all items for which the recommender system estimates high rating scores, i.e., "top items". For a recommender system model \mathcal{R} and k neighbors, a user u 's set of top items is given by $\mathcal{R}_{top}^k(u) = \arg \max_{i \in Q_u} \mathcal{R}^k(u, i)$, where Q_u are the items in u 's query set. In our case, we set $n = 10$.

Positivity-Popularity Correlation. To study whether higher estimated rating scores are given to popular items, we follow Kowald et al. [39] and correlate an item's popularity with its occurrences in users' sets of top items: $\text{ItemFreq}^+@k(i) = \sum_{u \in U} \mathbb{1}_{\mathcal{R}_{top}^k(u)}(i)$, where $\mathbb{1}_{\mathcal{R}_{top}^k(u)}(i)$ indicates whether item i is in user u 's set of top items $\mathcal{R}_{top}^k(u)$. Plus, an item i 's popularity is given by $\text{ItemPop}(i) = |U_i|/|U|$, where U is the set of all users and U_i are the users that rated i . We compute the Pearson correlation coefficient [6] between the two variables ItemFreq^+ and ItemPop to identify item popularity bias:

$$\text{PP-Corr}@k = \frac{\sigma_{\text{ItemFreq}^+@k, \text{ItemPop}@k}}{\sigma_{\text{ItemFreq}^+@k} \cdot \sigma_{\text{ItemPop}@k}}, \quad (18)$$

Table 3. Descriptive Statistics of the Five Datasets

Dataset	Domain	Rating range	$ U $	$ I $	$ R $	$ R / U $	$ U / I $	Density
ML 1M	Movies	{1...5}	6,040	3,706	1,000,209	165.60	1.6298	4.47%
Douban	Movies	{1...5}	2,509	39,576	893,575	356.15	0.0634	0.90%
LastFM	Music	{1...1,000}	3,000	352,805	1,755,361	585.12	0.0085	0.17%
Ciao	Movies	{1...5}	7,375	105,096	282,619	38.32	0.0702	0.04%
Goodreads	Books	{1...5}	20,000	508,696	2,569,177	128.46	0.0394	0.03%

$|U|$ is the number of users, $|I|$ is the number of items, $|R|$ is the number of ratings, $|R|/|U|$ is the ratings-to-users ratio, $|U|/|I|$ is the users-to-items ratio, and Density is given by $|R|/(|U| \cdot |I|)$.

where $\sigma_{\text{ItemFreq}^+@k, \text{ItemPop}@k}$ is the sample covariance between $\text{ItemFreq}^+@k$ and $\text{ItemPop}@k$. The sample standard deviations are given by $\sigma_{\text{ItemFreq}^+@k}$ and $\sigma_{\text{ItemPop}@k}$.

Item Coverage. In addition to evaluating the correlation between an item’s estimated rating score and its popularity, we measure the fraction of items that are a top item for at least one user. For this, we use the Item Coverage metric [31] given by

$$\text{Coverage}@k = \frac{1}{|I|} \left| \bigcup_{u \in U} \mathcal{R}_{top}^k(u) \right|, \quad (19)$$

where k is the number of neighbors, I is the set of items, U is the set of users, and $\mathcal{R}_{top}^k(u)$ is the set of top items for user u . This way, we can test whether parts of the item catalog always receive low estimated rating scores. We use PP-Corr@ k and Coverage@ k to evaluate ReuseKNN_{DP} . Additionally, we use these metrics to evaluate UserKNN to explore the impact of DP [21].

5.3 Datasets

In this work, we conduct experiments on five different datasets: *MovieLens 1M* (ML 1M) [28], *Douban* [34], *LastFM User Groups* (LastFM) [39], *Ciao* [26], and *Goodreads* [57, 58].

All five datasets exhibit different properties, as illustrated in Table 3. For example, the movie rating dataset *ML 1M* (integer ratings in {1...5}) is the densest dataset. Similarly, *Douban* (integer ratings in {1...5}) and *Ciao* (integer ratings in {1...5}) are movie rating datasets. Moreover, in *Ciao*, users have the smallest number of ratings per user (i.e., $|R|/|U|$) on average. *LastFM* includes implicit feedback data (i.e., listening counts) from the online music streaming service Last.fm. However, in this dataset, Kowald et al. [39] transfer the implicit feedback to decimal ratings in {1...1,000}. Plus, users have the largest number of ratings per users. The book rating dataset *Goodreads* (integer ratings in {1...5}), for which we use a random sample of 20,000 users, is the largest and least dense dataset.

Overall, the datasets cover (i) the movie, music, and book domain; (ii) implicit and explicit feedback; and (iii) different descriptive statistics.

5.4 Evaluation Protocol and Statistical Tests

We perform all experiments using 5-fold cross-validation, and randomly split all folds into 80% training sets R^{train} and 20% test sets R^{test} . The ratings in R^{train} are used to train the recommendation algorithms, and the ratings in R^{test} represent the rating queries used for evaluation. Also, we test the statistical significance of our results. Specifically, after close inspection of our results, we resort to the Mann-Whitney-U-Test. For the query-based metrics Neighbors@ q and CoRatings@ q , we evaluate significance for all rating queries $q \in [2; 100]$ when utilizing $k = 10$ neighbors. For other metrics, i.e., MAE@ k , PrivacyRisk@ k , PP-Corr@ k , and Coverage@ k , we evaluate significance after all queries have been processed by the recommender system. Again,

here, we utilize $k = 10$ neighbors to generate recommendations. Importantly, throughout this work, we only report statistical significance if we observe significance for each of the five folds.

5.5 Parameter Settings

The proposed approach relies on two adjustable hyperparameters: (i) the number of neighbors k used in the recommendation process and (ii) the data usage threshold τ . To test the performance of *ReuseKNN* and *ReuseKNN_{DP}* for different values of k , we vary $k \in \{5, 10, 15, 20, 25, 30\}$. Plus, we set τ to the approximate starting value of the tail of *UserKNN*'s data usage distribution DataUsage@k, which is given by its maximal second derivative (see Figure 1). This way, we assume that only the tail's small privacy risk (as a result of the rare data usage) is tolerable and give an example of how τ can be defined by the recommender system provider. Also, τ is the same for all users. This leads to the following τ values for $k = 10$: 92.89 (ML 1M), 91.54 (Douban), 104.32 (LastFM), 95.79 (Ciao), and 94.90 (Goodreads). For the similarity function sim , we use cosine similarity.

6 RESULTS AND DISCUSSION

We structure our results into two parts: (i) neighborhood reuse only (*ReuseKNN*), and (ii) neighborhood reuse with DP (*ReuseKNN_{DP}*).

6.1 ReuseKNN

In this section, we present our evaluation results for *ReuseKNN*, i.e., neighborhood reuse only.

6.1.1 Neighborhood Reuse. As the first step in this evaluation stage, neighborhood reuse only, we investigate the neighborhoods generated by *ReuseKNN*. Specifically, we compare our neighborhood reuse strategies to our *UserKNN* baseline with respect to the neighborhood size and the number of co-ratings with the target user. Moreover, we test for statistical significant differences to *UserKNN* after multiple rating queries, i.e., for all $q \in [2; 100]$.

We investigate the average size of target users' neighborhood after q rating queries for a model with $k = 10$ neighbors in Figure 3. For all of our five datasets, the size of a user's neighborhood increases more strongly for traditional *UserKNN* than for our neighborhood reuse strategies. For ML 1M, Douban, LastFM, and Goodreads, a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) shows that all our neighborhood reuse strategies yield significantly smaller neighborhoods than traditional *UserKNN* for $q \in [2; 100]$ rating queries. This means that *ReuseKNN* can already reuse neighbors after an initial neighborhood is generated for the very first rating query.

However, for Ciao, multiple initial rating queries are needed to generate reusable neighborhoods. Our neighborhood reuse strategies tend to yield significantly smaller neighborhoods only for a few rating queries. For *Gain*, we do not observe significant differences. We attribute this to the on average small user profiles in Ciao (see Table 3). Reusable neighbors are scarce and, thus, *ReuseKNN* cannot reduce the neighborhood size as effectively as in the case of the other datasets.

In addition to the neighborhood size, we also investigate the number of co-rated items between the target user and its neighbors after querying q rating queries (see Figure 4). Note that, as before, the statistical significance is evaluated after multiple rating queries, i.e., for all $q \in [2; 100]$. For all of our five datasets, our neighborhood reuse strategies can substantially increase the number of co-ratings over traditional *UserKNN*. A one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) reveals that our neighborhood reuse strategies generate neighborhoods with significantly more co-ratings with the target user than *UserKNN* for $q \in [2; 100]$ rating queries. This indicates that *ReuseKNN* generates neighborhoods with fewer neighbors that have more co-ratings with the target user than in the case of traditional *UserKNN*, which can foster recommendation accuracy [2, 16].

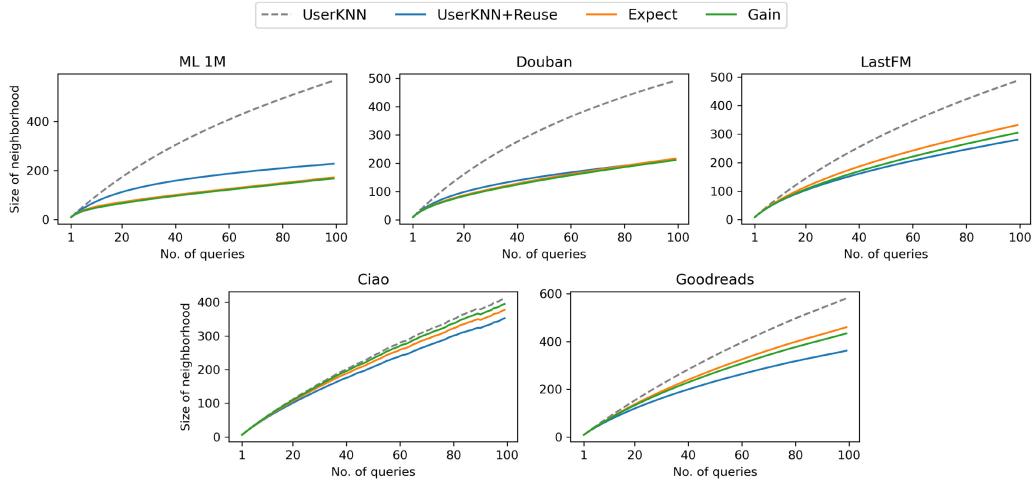


Fig. 3. Average number of neighbors per target user after q rating queries. Our neighborhood reuse strategies utilized in *ReuseKNN*, i.e., *Expect* and *Gain*, generate smaller neighborhoods than *UserKNN*.

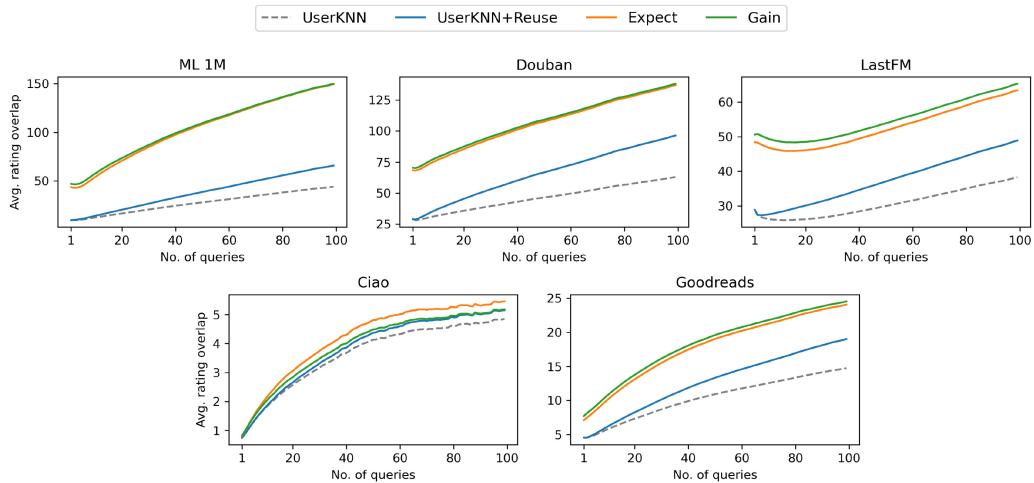


Fig. 4. Avg. number of co-rated items between the target user and its neighbors. Our neighborhood reuse strategies for *ReuseKNN*, i.e., *Expect* and *Gain*, generate neighborhoods, in which the neighbors' rated items overlap more with the target users' than in the case of *UserKNN*. With this, neighbors are beneficial for generating accurate recommendations.

However, for Ciao, our neighborhood reuse strategies tend to generate neighborhoods with significantly more co-ratings for only a few rating queries. As in our neighborhood size experiment, we attribute this to the small user profiles in Ciao, which makes neighborhood reuse less effective due to the scarcity of reusable neighbors.

6.1.2 Accuracy. Next, we compare *ReuseKNN* with traditional *UserKNN* in terms of recommendation accuracy (see Figure 5). Specifically, we test for statistically significant differences between our neighborhood reuse strategies and the *UserKNN* baseline.

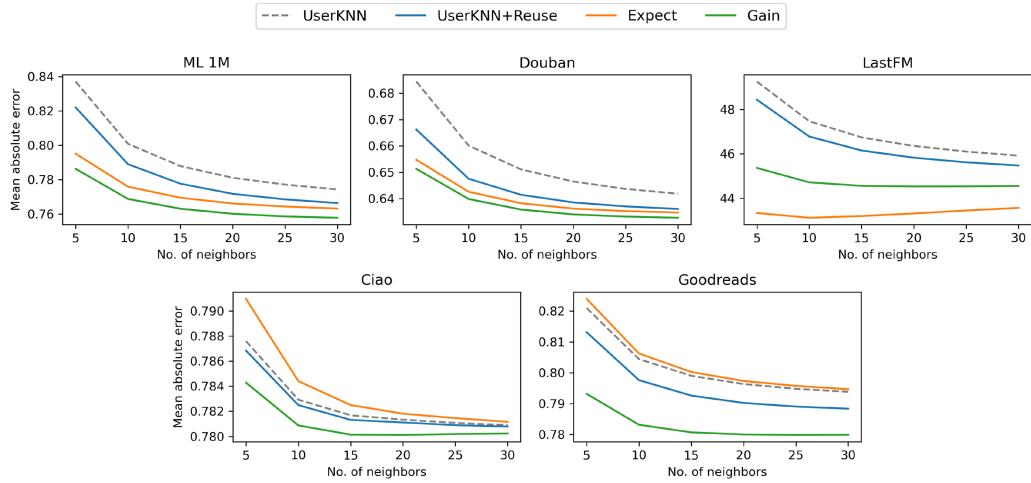


Fig. 5. Comparison of the recommendation accuracy between *ReuseKNN* and *UserKNN*. *ReuseKNN*'s neighborhood reuse strategies generate more accurate recommendations than *UserKNN*. For sparse datasets (i.e., Ciao and Goodreads), personalized neighborhood reuse (i.e., *Gain*) works better. In contrast, unpersonalized neighborhood reuse (i.e., *Expect*) works better for datasets, in which neighbors are scarce (i.e., LastFM).

We find that our neighborhood reuse strategies can generate more accurate recommendations than *UserKNN*. This shows that reusing neighbors that have already been used in the past can also lead to meaningful (accurate) recommendations in the future. Specifically, for ML 1M, Douban, and LastFM, a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) indicates that our neighborhood reuse strategies significantly increase recommendation accuracy for a model with $k = 10$ neighbors. Due to personalization, *Gain* performs best across most datasets.

For LastFM, unpersonalized neighborhood reuse (i.e., *Expect*) outperforms personalized neighborhood reuse (i.e., *Gain*). We attribute this to LastFM's small users-to-items ratio as compared with the other datasets (see Table 3), which makes it hard to identify neighbors, similar to an item-cold start scenario [52]. Concretely, in the case of personalized neighborhood reuse, selecting reusable neighbors for a specific target user reduces the pool of potential neighbors per item to a personalized subset and leads to a worse performance compared with unpersonalized neighborhood reuse. In contrast, unpersonalized neighborhood reuse allows using the entire pool of potential neighbors and, thus, achieves a higher accuracy for LastFM.

In the case of our least dense datasets Ciao and Goodreads, we observe that our personalized neighborhood reuse strategy *Gain* can handle these datasets better than our unpersonalized neighborhood reuse strategy *Expect*. *Gain* selects neighbors whose rating data could have been used by the target user in the past (see Equation (6)). This way, *Gain* creates a neighborhood for a given target user with sufficient rating data even in sparse datasets.

Plus, we highlight that *Gain* significantly increases recommendation accuracy for Goodreads despite the dataset's low density. In the case of Ciao, a two-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) reveals no significant differences between our neighborhood reuse strategies and *UserKNN* for $k = 10$, which suggests that all our neighborhood reuse strategies can preserve recommendation accuracy. As shown in our previous experiments (see Section 6.1.1), neighborhood reuse is less effective for Ciao due to the small user profiles. Thus, it makes sense that for Ciao, the recommendation accuracy cannot be improved as effectively as for the remaining datasets.

Table 4. Percentage of Vulnerable Users for a Model with $k = 10$ Neighbors

Method	ML 1M	Douban	LastFM	Ciao	Goodreads
UserKNN	80.39%	96.68%	99.89%	8.02%	65.00%
UserKNN+Reuse	84.64%	87.37%	98.90%	7.91%	52.29%
Expect	24.13%	34.40%	68.20%	7.88%	29.12%
Gain	25.09%	37.43%	80.28%	8.19%	40.51%

Best results, i.e., lowest values, are in **bold**. For all datasets, *ReuseKNN*'s *Expect* neighborhood reuse strategy leads to fewer vulnerable users than *UserKNN*. For Ciao, our neighborhood reuse strategies can achieve only minor improvements, as already *UserKNN* yields a small percentage of vulnerable users.

6.1.3 Percentage of Vulnerable Users. In Section 6.1.1, we found that neighborhood reuse can significantly reduce the number of neighbors that are utilized in the recommendation process. Now, however, we analyze how many neighbors are utilized for more than τ rating queries (i.e., the usage of their data exceeds threshold τ) and, thus, need to be protected with DP (see Table 4). Specifically, we compare our neighborhood reuse strategies to the *UserKNN* baseline.

For all of our five datasets, our neighborhood reuse strategies lead to less vulnerable users than traditional *UserKNN*. Especially, *Expect* shows the best (i.e., lowest) percentage of vulnerable users. For example, for the ML 1M dataset, *UserKNN* leads to 80.39% of users that are vulnerable, since their data usage exceeds threshold $\tau = 92.89$ (see Section 5.5), whereas *Expect* leads to only 24.13% vulnerable users and, thus, fewer users need to be protected with DP.

For Ciao, our neighborhood reuse strategies achieve only minor improvements over *UserKNN*. The reason is that *UserKNN* already yields a small percentage of vulnerable users and, as such, *ReuseKNN* leads to only small improvements. Additionally, our previous findings show that the effect of neighborhood reuse on Ciao is smaller than on the remaining datasets due to the small average user profile size (see Table 3). This leads to a lack of reusable neighbors and, thus, also limits the effect that neighborhood reuse has on the percentage of vulnerable users.

6.1.4 Summary. Overall, we find that through neighborhood reuse, *ReuseKNN* can significantly reduce the size of target users' neighborhoods as compared with traditional *UserKNN*. Despite the much smaller neighborhoods, *ReuseKNN* identifies neighbors that have many more co-rated items with the target user than in the case of *UserKNN*. As related work suggests, these neighbors are more "reliable" and can be crucial for recommendation accuracy [2, 16].

Based on the much smaller but more reliable neighborhoods, *ReuseKNN* can provide significantly higher recommendation accuracy than traditional *UserKNN*. For sparse datasets, personalized neighborhood reuse seems to be a better solution than unpersonalized neighborhood reuse.

Plus, *ReuseKNN* can substantially reduce the percentage of vulnerable users, and in general, our *Expect* neighborhood reuse method yields the fewest vulnerable users.

6.2 ReuseKNN_{DP}

Next, we present our results on *ReuseKNN*_{DP}, i.e., neighborhood reuse with DP.

6.2.1 Accuracy. First and foremost, we note that in our experiments without DP (see Figure 5), *UserKNN* could be outperformed by *ReuseKNN*. In our experiments with DP, however (see Figure 6), it is apparent that all evaluated DP methods do not reach the accuracy of non-DP *UserKNN*. This means that in general, due to DP, drops in recommendation accuracy have to be expected. However, we will investigate next whether *ReuseKNN*_{DP} can make this accuracy drop less severe compared with using the baselines. In detail, we compare our neighborhood reuse strategies to the *UserKNN*_{DP} baseline and test for statistically significant differences.

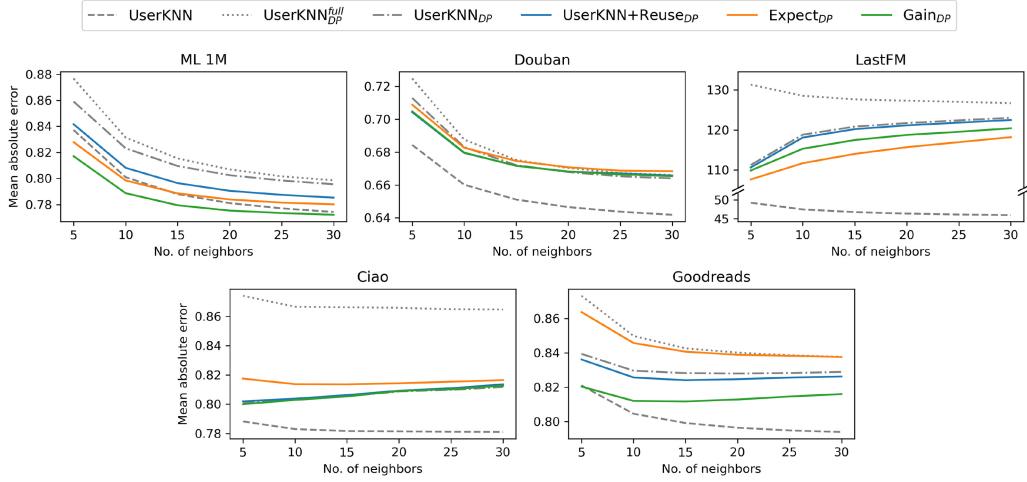


Fig. 6. Comparison of the recommendation accuracy between $ReuseKNN_{DP}$ and $UserKNN_{DP}$. We find that $ReuseKNN_{DP}$'s neighborhood reuse strategies, $Expect_{DP}$ and $Gain_{DP}$, can preserve or even improve recommendation accuracy in terms of lower MAE. This shows that reducing the number of users to which DP has to be applied can help to increase recommendation accuracy.

Furthermore, we incorporate $UserKNN$ without DP and $UserKNN_{DP}^{full}$ as additional baselines for our experiments.

In general, for our neighborhood reuse strategies, DP does not cause an accuracy drop as severe as in case of $UserKNN_{DP}$ (see Figure 6). Plus, as expected, $UserKNN_{DP}^{full}$ performs worst due to the randomness that is added via DP to the rating data of all users. This shows that our neighborhood reuse concept helps to generate accurate recommendations in differentially private KNN-based recommender systems. For ML 1M and LastFM, a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) indicates that our neighborhood reuse strategies significantly increase recommendation accuracy over $UserKNN_{DP}$ for a model with $k = 10$ neighbors. Additionally, for ML 1M, $Gain_{DP}$ performs better than our non-DP baseline $UserKNN$.

Moreover, we observe that LastFM is highly sensitive to the incorporation of DP, since the mean absolute error magnitudes differ substantially between our non-DP experiment in Figure 5 and our DP experiment in Figure 6. In line with our previous results on non-DP $ReuseKNN$, $ReuseKNN_{DP}$'s unpersonalized neighborhood reuse strategy $Expect_{DP}$ also cannot increase recommendation accuracy for Ciao and Goodreads, which are our two sparsest datasets. However, our personalized neighborhood reuse strategy $Gain_{DP}$ generates recommendations with significantly higher accuracy for Goodreads. For Ciao, no significant differences are found according to a two-tailed Mann-Whitney-U-Test ($\alpha = 0.01$). Thus, $Gain_{DP}$ can preserve recommendation accuracy.

For Douban, we observe no significant differences between our neighborhood reuse strategies and $UserKNN_{DP}$. We found empirically that for Douban, $UserKNN_{DP}$ and $ReuseKNN_{DP}$ utilize more rating data from vulnerable users than in the case of our remaining datasets. Thus, we measure the fraction of rating data; each user contributes to the dataset, i.e., $|R_u|/|R|$, where R are all users' ratings and R_u are user u 's ratings. We find that for Douban, the 5% of users with the largest user profiles contribute substantially more ratings to the dataset than for our other datasets: 0.0008 (ML 1M), 0.0022 (Douban), 0.0012 (LastFM), 0.0009 (Ciao), and 0.0003 (Goodreads). This suggests that in the case of Douban, the recommendation process more often utilizes these users due to their abundance of rating data. This, however, makes these users more vulnerable. Therefore, we

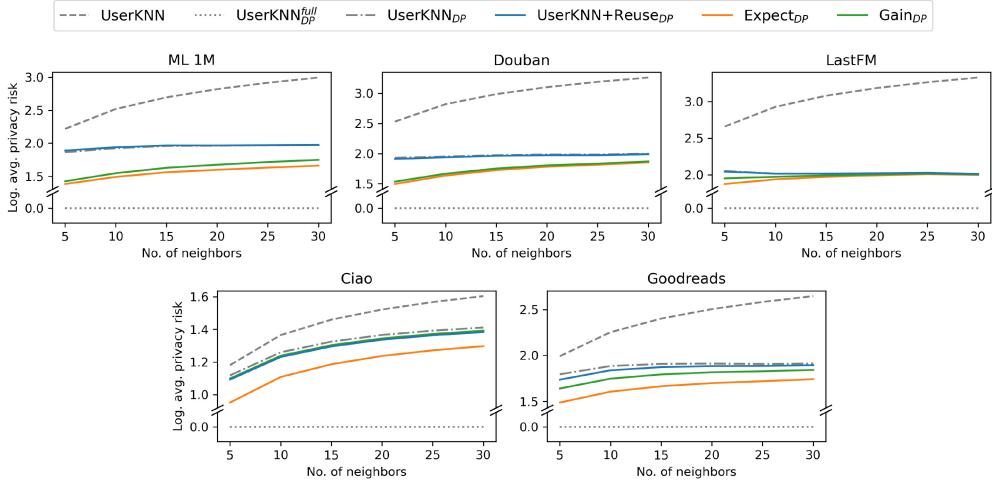


Fig. 7. Logarithm (base 10) of the privacy risk averaged over all users. $ReuseKNN_{DP}$'s neighborhood reuse strategies yield lower privacy risk than $UserKNN_{DP}$. This is due to the fact that $ReuseKNN_{DP}$ reduces the percentage of users with a privacy risk of τ (i.e., vulnerable) and simultaneously decreases the privacy risk of the remaining users (i.e., secure). Overall, we find that our unpersonalized neighborhood reuse strategy $Expect_{DP}$ achieves the best user privacy, i.e., the lowest privacy risk.

suppose that this strong utilization of DP-protected rating data from vulnerable users leads to no significant differences in accuracy between $UserKNN_{DP}$ and $ReuseKNN_{DP}$.

For Douban, we additionally compare $ReuseKNN_{DP}$ to $UserKNN_{DP}^{full}$. Our results suggest that our personalized reuse strategy $Gain_{DP}$ generates recommendations with significantly higher accuracy, where $Expect_{DP}$ shows no significant differences. Thus, all our neighborhood reuse strategies can preserve recommendation accuracy for this dataset.

6.2.2 Privacy Risk. In $ReuseKNN_{DP}$, vulnerable users with high data usage are protected with DP and as such, their privacy risk is set to threshold τ . Moreover, secure users' privacy risk is also reduced since they are rarely exploited as neighbors in the recommendation process, i.e., low data usage (see Figure 1). Specifically, we compare our neighborhood reuse strategies to $UserKNN_{DP}$ and test for statistically significant differences. Furthermore, we use $UserKNN$ without DP and $Full_{DP}$ as additional baselines.

We visualize the privacy risk of $ReuseKNN_{DP}$ and our three baselines $UserKNN$, $UserKNN_{DP}$, and $UserKNN_{DP}^{full}$ in Figure 7. We find that our neighborhood reuse strategies combined with DP can improve user privacy over $UserKNN_{DP}$. Specifically, a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) reveals that for our neighborhood reuse strategies on all datasets and for $k = 10$, users have significantly less privacy risk than in $UserKNN_{DP}$.

However, for LastFM, this privacy improvement is smaller than for the other datasets. Due to the large percentage of vulnerable users for all approaches (see Table 4), most users' privacy risk is set to τ due to the application of DP. Thus, the small percentage of secure users is insufficient to reduce the average privacy risk via neighborhood reuse in the case of LastFM.

Across all datasets, we observe that our unpersonalized neighborhood reuse strategy $Expect_{DP}$ yields the best (lowest) privacy risk. This finding is in line with our previous results in Table 4, which show that $Expect_{DP}$ performs best with respect to minimizing the percentage of vulnerable users. Thus, only a few users have a privacy risk of τ , and the high number of secure users enables

a drastic reduction of the average privacy risk. For example, the average privacy risk of secure users for a model with $k = 10$ neighbors for Expect_{DP} is 11.45 for ML 1M, 18.34 for Douban, 49.92 for LastFM, 15.29 for Ciao, and 18.99 for Goodreads compared with the privacy risk of secure users for UserKNN_{DP} , which is 50.83 for ML 1M, 62.13 for Douban, 73.42 for LastFM, 21.76 for Ciao, and 41.13 for Goodreads. Additionally, a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) reveals that for ML 1M, Douban, Ciao, and Goodreads, these differences are significant. Thus, for secure users, Expect_{DP} yields a substantially smaller privacy risk than UserKNN_{DP} .

6.2.3 Item Popularity Bias. We test for item popularity bias in ReuseKNN_{DP} 's recommendations via comparing ReuseKNN_{DP} to our UserKNN_{DP} baseline with respect to two metrics: Positivity-Popularity Correlation (PP-Corr) and Item Coverage (Coverage). Plus, we use UserKNN without DP and $\text{UserKNN}_{DP}^{full}$ as additional baselines. Moreover, in the case of PP-Corr, we test for statistically significant differences between our neighborhood reuse strategies and UserKNN_{DP} (see Table 3). First and foremost, for ML 1M, Douban, LastFM, and Ciao, the non-DP baseline UserKNN yields lower PP-Corr values than all remaining methods that use DP. Similarly, applying DP to only vulnerable users yields lower PP-Corr values than applying DP to all users in the case of ML 1M, Douban, Ciao, and Goodreads. This fits well to related research [21] arguing that popularity bias can arise due to the recommender system's inability to personalize recommendations when DP is applied.

However, ReuseKNN_{DP} can make the impact of DP on popularity bias less severe, since our neighborhood reuse strategies yield a lower PP-Corr than the DP baseline UserKNN_{DP} . No notable differences can be observed for Ciao only. We investigate this in more detail and find that the neighbors identified by ReuseKNN_{DP} rated more distinct items than the neighbors identified by UserKNN_{DP} . As shown by related work on item popularity bias in recommender systems (e.g., [1, 39]), users with a larger user profile size tend to consume less popular items, which leads to less popularity bias. Due to the small number of ratings per user in Ciao (see Table 3), which is similar to a user cold-start setting [40], no noteworthy effects on popularity bias can be observed.

In addition to PP-Corr, we also evaluate Coverage, i.e., the percentage of items from the entire item catalog that occur within users' sets of top items. In general, $\text{UserKNN}_{DP}^{full}$ tends to give the highest item coverage and non-DP UserKNN yields the lowest item coverage. This makes sense since $\text{UserKNN}_{DP}^{full}$ protects all rating data with DP and, thus, the estimated rating scores are more random than for the remaining approaches. This leads to more randomized recommendations, and, therefore, to high item coverage [22]. These randomized recommendations also lead to the fact that, in Table 5, ReuseKNN_{DP} cannot reach the item coverage of $\text{UserKNN}_{DP}^{full}$. However, more randomized recommendations lead to poorer accuracy than our previous results in Figure 6 show.

Our neighborhood reuse strategies cover fewer items than UserKNN_{DP} only in the case of LastFM. We underline that these item coverage values are negatively correlated with our accuracy results in Figure 6. This indicates that for LastFM, there is a trade-off between recommendation accuracy and item coverage similar to the well-known trade-off between precision and recall [8].

6.2.4 Summary. Overall, our results are in line with the previously presented results for our non-DP ReuseKNN . Through neighborhood reuse and, thus, reducing the number of users that need to be protected with DP, recommendation accuracy can be preserved and, in many cases, even significantly improved over UserKNN_{DP} .

Also, our neighborhood reuse strategies used in ReuseKNN_{DP} lead to significantly smaller privacy risk than UserKNN_{DP} . In particular, unpersonalized neighborhood reuse (i.e., Expect_{DP}) performs best in increasing user privacy. This shows that the combination of neighborhood reuse and DP provides higher privacy than UserKNN_{DP} .

Table 5. PP-Corr and Item Coverage for a Model with $k = 10$ Neighbors

	ML 1M		Douban		LastFM		Ciao		Goodreads	
	PP-Corr	Coverage	PP-Corr	Coverage	PP-Corr	Coverage	PP-Corr	Coverage	PP-Corr	Coverage
UserKNN	0.8405	87.94%	0.6780	23.50%	0.7339	6.11%	0.9755	63.19%	0.9318	29.56%
UserKNN _{DP}	0.8742	88.77%	0.7589	26.55%	0.8625	15.54%	0.9758	64.03%	0.9409	31.59%
UserKNN _{DP} ^{full}	0.8800	89.53%	0.7675	27.65%	0.8597	15.86%	0.9778	66.72%	0.9523	34.13%
UserKNN+Reuse _{DP}	0.8750	88.37%	0.7523	27.67%	0.8779	15.46%	0.9759	64.26%	0.9407	31.74%
Expect _{DP}	0.8688	88.83%	**0.7400	28.75%	0.8773	14.32%	0.9767	64.58%	**0.9317	34.69%
Gain _{DP}	0.8725	88.07%	**0.7428	28.61%	0.8621	14.77%	0.9769	64.01%	0.9454	31.46%

Best results, i.e., highest for PP-Corr and lowest for Coverage, are in **bold**. For PP-Corr, a z-Test [32] shows, with ** ($\alpha = 0.01$) that our neighborhood reuse strategies as utilized in *ReuseKNN_{DP}* lead to estimated rating scores that are significantly less correlated with item popularity than in case of *UserKNN_{DP}*. With respect to item coverage, especially *Expect_{DP}* can cover a larger percentage of the item catalog than *UserKNN_{DP}*. Overall, our results suggest that *ReuseKNN_{DP}* does not increase item popularity bias over *UserKNN_{DP}*.

Table 6. Mean Absolute Error and Average Privacy Risk Values for our Neighborhood Reuse Strategies Used in *ReuseKNN_{DP}*, i.e., *Expect_{DP}* and *Gain_{DP}* and for the *UserKNN_{DP}* Baseline ($k = 10$)

	ML 1M		Douban		LastFM		Ciao		Goodreads	
	MAE	Privacy R.	MAE	Privacy R.	MAE	Privacy R.	MAE	Privacy R.	MAE	Privacy R.
UserKNN	0.80	330.77	0.66	665.17	47.46	844.94	0.78	35.21	0.80	182.26
UserKNN _{DP}	0.82	84.39	0.68	89.86	118.80	103.77	0.81	27.61	0.83	75.71
UserKNN _{DP} ^{full}	0.83	0.00	0.69	0.00	128.41	0.00	0.87	0.00	0.85	0.00
UserKNN+Reuse _{DP}	0.81	87.16	0.68	87.16	118.13	103.56	0.81	26.54	0.83	68.35
Expect _{DP}	**0.80	**31.03	0.68	**43.25	**11.78	**86.81	0.82	**21.53	0.85	**40.95
Gain _{DP}	**0.79	**35.30	0.68	**46.57	**115.31	**93.95	0.81	**26.74	**0.81	**55.90

Also, we perform a one-tailed Mann-Whitney-U-Test ($\alpha = 0.01$) and mark (with **) significantly better (i.e., Lower) values than *UserKNN_{DP}*. Overall, personalized neighborhood reuse (i.e., *Gain_{DP}*) yields the best accuracy and unpersonalized neighborhood reuse (i.e., *Expect_{DP}*) gives the lowest privacy risk. For Douban and LastFM, *Expect_{DP}* is well-suited as it yields the highest accuracy and lowest privacy risk. For the remaining datasets, all neighborhood reuse strategies provide a less serious accuracy-privacy trade-off than *UserKNN_{DP}*.

In addition, we find that for *ReuseKNN_{DP}*, high estimated rating scores are weaker correlated to item popularity than in the case of *UserKNN_{DP}* and that *ReuseKNN_{DP}* can estimate high rating scores for more items than *UserKNN_{DP}*. Thus, *ReuseKNN_{DP}* does not increase item popularity bias.

6.3 Discussion

We provide a condensed summary of experimental results (see Table 6) for all evaluated approaches and all five datasets. Specifically, we present the accuracy (i.e., MAE@ k) and average privacy risk (i.e., PrivacyRisk@ k) values for a model with $k = 10$ neighbors.

Overall, non-DP *UserKNN* results in low MAE but high privacy risk values. This shows that approaches without DP sacrifice a user's privacy for recommendation accuracy. However, our neighborhood reuse strategies with DP provide a less serious trade-off between recommendation accuracy and privacy. Thus, in the following, we briefly discuss advantages and disadvantages of our neighborhood reuse strategies for all five datasets.

Across our neighborhood reuse strategies that are utilized in *ReuseKNN_{DP}*, in general, personalized neighborhood reuse (*Gain_{DP}*) provides the best recommendation accuracy. Plus, unpersonalized neighborhood reuse (*Expect_{DP}*) yields the lowest privacy risk. For Douban and LastFM, *Expect_{DP}* performs best in both accuracy and privacy risk. Thus, in this case, *Expect_{DP}* is well suited to provide accurate and private recommendations. For ML 1M, Ciao, and Goodreads, no neighborhood reuse strategy provides the best result in both evaluation criteria. Thus, it depends on the recommender system service provider to decide what strategy could be utilized.

6.4 Additional Considerations and Experiments

While our experiments reported so far considered a rating prediction task as motivated by our problem statement in Section 3 (accordingly, we measured accuracy using the MAE [51]), we perform additional experiments with regards to a ranking-based recommendation scenario and a neural-based recommender system. Due to space limitations, the results of these are detailed in the appendices of this article. First, we model a ranking-based recommendation scenario, which is very common today. Accordingly, we perform experiments using a ranking-based evaluation metric, nDCG [35], and report results in Appendix B. Given the widespread adoption of deep learning techniques in the latest recommender systems, we also incorporate neighborhood reuse into a popular neural-based approach, neural collaborative filtering (NeuCF) [29]. The approach and results are detailed in Appendix C.

Overall, our additional experiments reveal the same pattern of results as discussed above. That is, the combination of neighborhood reuse and DP can provide a better trade-off between accuracy and privacy than recommendation methods without neighborhood reuse. This shows the generalizability of the neighborhood reuse principle for other evaluation scenarios and recommendation algorithms.

7 CONCLUSION

In this work, we investigate the efficacy of neighborhood reuse for differentially private KNN-based recommendations. We discuss the proposed approach in a two-stage evaluation procedure: (i) neighborhood reuse only, *ReuseKNN*, to distill the impact of neighborhood reuse on recommendation accuracy and on the percentage of users that need to be protected with differential privacy; and (ii) neighborhood reuse with differential privacy, *ReuseKNN_{DP}*, to investigate the practical benefit of neighborhood reuse for differentially private KNN-based recommendations. We find that *ReuseKNN* and *ReuseKNN_{DP}* can substantially reduce the number of users that need to be protected with DP while outperforming related approaches in terms of accuracy. Also, we highlight that *ReuseKNN_{DP}* effectively mitigates users' privacy risk, as most users are rarely exploited in the recommendation process. Our work illustrates how to address privacy risks in recommender systems through neighborhood reuse combined with DP.

Limitations. We recognize two limitations of the proposed approach. To quantify the privacy risk, we assume that all pieces of data are equally sensitive. In reality, disclosing a particular piece of information could pose a different level of privacy risk than disclosing another piece of information [38, 45]. Also, we focus on a neighborhood-based recommender system, specifically user-based KNN, instead of neural-based recommender systems. The latter are popular due to their ability to extract and exploit rich user and item representations for generating recommendations. However, traditional algorithms, such as user-based KNN, have been shown to perform well in a variety of real-world use cases [15]. Plus, neighborhood-based recommender systems have the advantage of providing justifiable recommendations and they incorporate new rating data of users efficiently without requiring a complete retraining of the whole model from scratch [16]. Nonetheless, we demonstrate in Appendix C that neighborhood reuse can be generalized to neural-based recommender systems, e.g., NeuCF [29].

Future Work. In this work, we evaluated the proposed approach using datasets of three different domains (movies, books, and music). Future work will consider additional, more sensitive domains, such as medicine, finance, insurance, and recruiting. We will also incorporate neighborhood reuse into other neural-based recommendation models, e.g., BERT4Rec [54]. Plus, we plan to study the impact of the proposed approach, i.e., neighborhood reuse and differential privacy, on individual users' preferences towards long-tail items, e.g., by using the dataset from our previous work on

fairness in music recommender systems [39]. Hence, our long-term plan is to investigate the interaction between privacy and fairness, two key aspects of trustworthy recommender systems.

MATERIALS

The Python-based implementation of our work is publicly available.¹ Also, we provide the source code for generating our sample of the Goodreads dataset. All remaining datasets are publicly available as well (see Section 5.3).

APPENDICES

A DETAILED DIFFERENTIAL PRIVACY ANALYSIS

Our differential privacy analysis relies on the fact that, even if the adversary is able to infer the rating used in the recommendation process, it is unaware whether this rating is the neighbor's real rating or was randomly generated by our m_{DP} mechanism. Formally:

$$\frac{Pr[\text{Adversary's assumption: Real rating} \mid \text{Truth: Real rating}]}{Pr[\text{Adversary's assumption: Real rating} \mid \text{Truth: Random rating}]} = \quad (20)$$

$$\frac{Pr[\text{Non-DP rating}] + Pr[\text{Real rating} \mid \text{DP rating}] \cdot Pr[\text{DP rating}]}{Pr[\text{Random rating} \mid \text{DP rating}] \cdot Pr[\text{DP rating}]} = \quad (21)$$

$$\frac{Pr[\text{Non-DP rating}]}{Pr[\text{Random rating} \mid \text{DP rating}] \cdot Pr[\text{DP rating}]} + \underbrace{\frac{Pr[\text{Real rating} \mid \text{DP rating}]}{Pr[\text{Random rating} \mid \text{DP rating}]}}_{m_{DP} \text{ mechanism}} = \quad (22)$$

$$\frac{1}{0.25} \cdot \frac{Pr[\text{Non-DP rating}]}{Pr[\text{DP rating}]} + \frac{0.75}{0.25} = \quad (23)$$

$$4 \cdot \frac{\frac{Pr[\text{PrivacyRisk}@k(u)]}{DataUsage@k(u)}}{DataUsage@k(u) - PrivacyRisk@k(u)} + 3 = \quad (24)$$

$$4 \cdot \frac{PrivacyRisk@k(u)}{DataUsage@k(u) - PrivacyRisk@k(u)} + 3 \leq e^\epsilon \quad (25)$$

which leads to a privacy parameter of

$$\epsilon = \ln \left(3 + 4 \cdot \frac{PrivacyRisk@k(u)}{DataUsage@k(u) - PrivacyRisk@k(u)} \right). \quad (26)$$

In the case of $UserKNN_{DP}^{full}$, all ratings of a user u are protected with DP and, therefore, $PrivacyRisk@k(u) = 0$, which leads to $\epsilon = \ln 3$. In the case of $UserKNN$, no DP is applied at all and, thus, computing ϵ is not possible since ϵ is part of the DP framework. Therefore, we set $\epsilon = \infty$. In the case of $UserKNN_{DP}$ and $ReuseKNN_{DP}$, DP is applied to the rating data of users, for which the usage of their data exceeds threshold τ . Assuming that u is vulnerable, then $DataUsage@k(u) > \tau$ and $PrivacyRisk@k(u) = \min[\tau, DataUsage@k(u)]$. Therefore, it follows that $0 < PrivacyRisk@k(u) < DataUsage@k(u)$. Varying $PrivacyRisk@k(u)$ within these boundaries yields:

$$\ln 3 < \ln \left(3 + 4 \cdot \frac{1}{DataUsage@k(u) - 1} \right) \leq \epsilon \leq \ln \left(3 + 4 \cdot (DataUsage@k(u) - 1) \right) < \infty. \quad (27)$$

¹<https://github.com/pmuellner/ReuseKNN>

This shows that $UserKNN_{DP}$ and $ReuseKNN_{DP}$ provide better privacy than $UserKNN$, but worse privacy than $UserKNN_{DP}^{full}$.

Moreover, via neighborhood reuse, $ReuseKNN_{DP}$ utilizes a vulnerable user u more often as neighbor (with DP-protected data) than $UserKNN_{DP}$ does. Also, note that the privacy risk of u is the same for $ReuseKNN_{DP}$ and $UserKNN_{DP}$. From these observations and Equation (26), we see that the ϵ value for $ReuseKNN_{DP}$ is smaller than the ϵ value for $UserKNN_{DP}$. Thus, for vulnerable users, our neighborhood reuse principle leads to $ReuseKNN_{DP}$ providing better privacy than $UserKNN_{DP}$.

B EVALUATION OF TOP-N RECOMMENDATIONS

In our article, we show that $ReuseKNN_{DP}$ can achieve better accuracy in terms of the rating prediction metric MAE than a traditional KNN recommender system with DP. In the following, we evaluate $ReuseKNN_{DP}$ in a top- n items recommendation setting via the ranking-aware metric $nDCG$ (Normalized Discounted Cumulative Gain) [35].

B.1 Evaluation Process

To generate a list of recommended items that can be evaluated via $nDCG$, we select the $n = 10$ items with the highest predicted rating score for a given target user u [51]. Formally, for a recommender system model \mathcal{R} and k neighbors, a user u 's top- n items are given by:

$$\mathcal{R}_{top}^k(u) = \arg \max_{i \in Q_u} \mathcal{R}^k(u, i) \quad (28)$$

where Q_u are the items in u 's query set. We consider items in the test set as relevant if their true rating exceeds the average rating in the training set of the given dataset.

B.2 Experiments

Our results reveal that $Expect_{DP}$ and $Gain_{DP}$ can yield higher $nDCG$ scores than $UserKNN_{DP}^{full}$ (see Figure 8). In the case of the ML 1M dataset, $Expect_{DP}$ and $Gain_{DP}$ can even outperform the non-DP baseline $UserKNN$. Especially $Gain_{DP}$ yields high $nDCG$ scores. Overall, this experiment validates

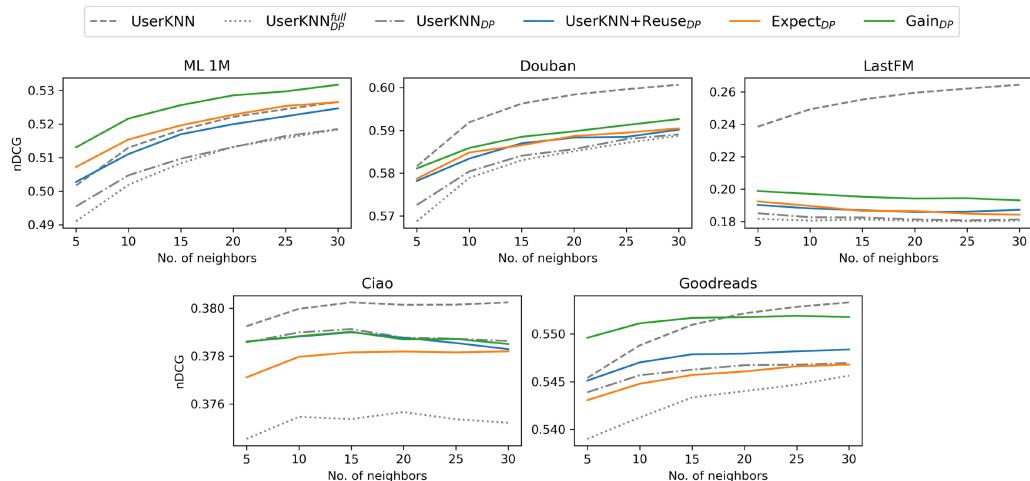


Fig. 8. $nDCG$ values of each user's top 10 items. The pattern matches our results reported in Section 6, i.e., $ReuseKNN_{DP}$ can yield better accuracy than $UserKNN_{DP}$. Also, especially personalized neighborhood reuse (i.e., $Gain_{DP}$) can preserve accuracy well.

the results of our rating prediction evaluation setting also in a top- n items recommendation setting.

C EVALUATION OF NEURAL-BASED RECOMMENDATIONS

This work considers rating data as input to the recommender system. However, recommender systems can also use more complex representations of users and items, i.e., embeddings as generated by neural network architectures. Therefore, in the following, we demonstrate the generalizability of our approach for neural-based recommendation methods.

C.1 Generation of Embeddings

To generate user and item embeddings, we rely on a simple approach inspired by the NeuCF [29] architecture. Specifically, for user u and item i , the predicted rating score $y_{u,i}$ is given by:

$$y_{u,i} = b + \text{ReLU}\left(wx_u W_u^T x_i W_i\right), \quad (29)$$

where x_u is the id of user u , x_i is the id of item i , the size of the embedding layer is $d = 16$, $W_u, W_i \in \mathbb{R}^d$, $w, b \in \mathbb{R}$, and ReLU is the activation function. We apply Adam [37] with a step size of $\alpha = 0.001$ to minimize the MAE between $y_{u,i}$ and the rating $r_{u,i}$. The parameters α and d are set to the values proposed in [29]. We train the network for 50 epochs and use a batch size of 128. We stop training if there is no improvement of the training objective for more than 10 epochs. After training, the user and item embeddings are given by $x_u W_u$ and $x_i W_i$ respectively.

C.2 Neural-Based Recommendations

For our neural-based variants of *UserKNN*–*NeuKNN* and *NeuKNN_{DP}*—we calculate the similarity between the target user and the candidate neighbors based on their user embeddings (see Equation (2)). For *NeuKNN+Reuse_{DP}*, i.e., an embedding-based variant of *ReuseKNN_{DP}*, we also use an embedding-based similarity. Plus, we employ a modified definition of *reusability* that measures the reusability of a candidate neighbor c based on the previous $t - 1$ rating queries of target user u :

$$\text{reusability}(c|u, i, t) = \sum_{j \in Q_u^{(t-1)}} \mathbb{1}_{N_{u,j}}(c) \cdot \text{sim}(i, j), \quad (30)$$

where $\mathbb{1}_{N_{u,j}}(c)$ is the indicator function of candidate neighbor c being in $N_{u,j}$. The item similarity sim is the cosine similarity between i 's and j 's item embeddings. Therefore, $\text{reusability}(c|u, i, t)$ is the summed-up item similarity between the target item i and all items $j \in Q_u^{(t-1)}$ (i.e., the previous $t - 1$ rating queries of u) for which c has been used as neighbor.

C.3 Experiments

In our experiments, we perform evaluation according to the following procedure: First, we randomly split the dataset into 5 equally sized subsets: $D_{1 \leq i \leq 5}$. We select D_1 and equally partition it into the validation data that is used for validating the user and item embeddings and the test data that is used for evaluating the recommendations. The remaining data, $\bigcup_{2 \leq i \leq 5} D_i$, is used to train the user and item embeddings and to generate recommendations. Next, we select D_i and repeat this procedure for all $D_{2 \leq i \leq 5}$. Eventually, we compute the mean of our evaluation results.

Accuracy. For all datasets, *NeuKNN+Reuse_{DP}* outperforms our baseline *NeuKNN_{DP}^{full}* that applies DP to all users (see Figure 9). For completeness, we also visualize *NeuKNN* that does not apply DP at all and, thus, yields higher accuracy than both DP-based methods. Overall, the result for our embedding-based methods *NeuKNN_{DP}^{full}* and *NeuKNN+Reuse_{DP}* are in line with the results of

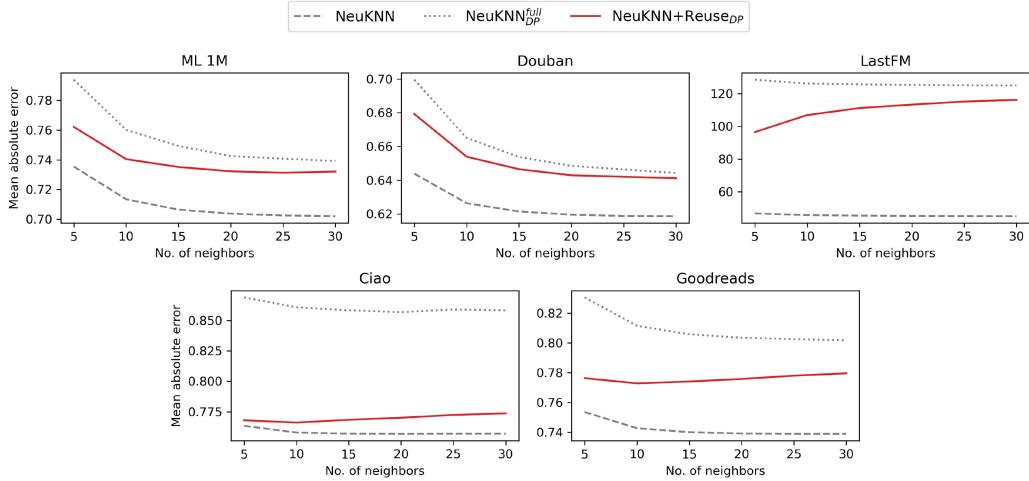


Fig. 9. Mean absolute error of our neural-based KNN recommender system variants. Our results indicate that combining neighborhood reuse with DP (i.e., $NeuKNN+Reuse_{DP}$) yields better accuracy (lower MAE) than neural-based methods that apply DP without neighborhood reuse (i.e., $NeuKNN_{DP}^{full}$).

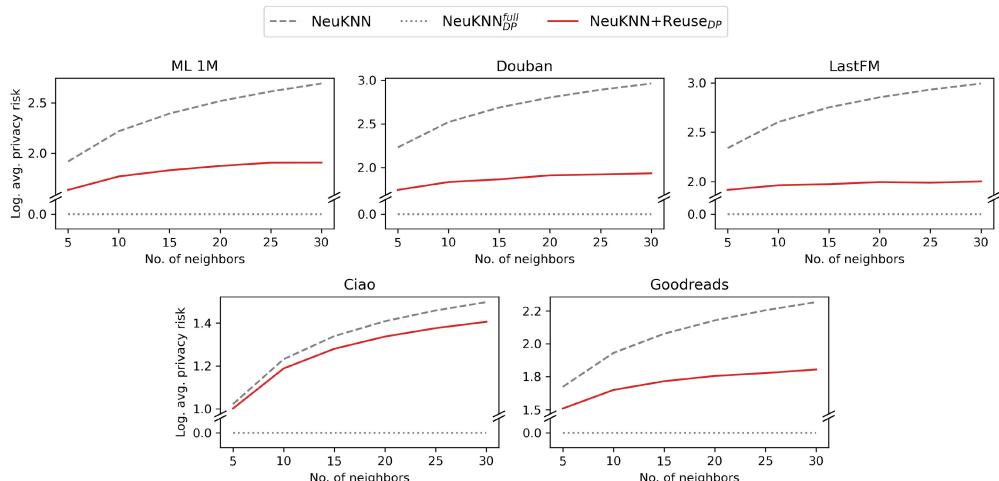


Fig. 10. Logarithmic (base 10) average privacy risk of our neural-based KNN recommender system variants. Via combining neighborhood reuse and DP, $NeuKNN+Reuse_{DP}$ decreases the users' average privacy risk compared with neural-based methods that do not apply DP (i.e., $NeuKNN$).

our rating-based methods, i.e., that the combination of neighborhood reuse and DP yields better accuracy on all five investigated datasets than traditional DP-based methods.

Privacy. Our baseline $NeuKNN$ without DP yields the worst privacy risk, whereas $NeuKNN_{DP}^{full}$ yields a privacy risk of zero since all users are protected with DP (see Figure 10). $NeuKNN+Reuse_{DP}$ protects only vulnerable users with DP; in this way, its privacy risk lies between our two baselines. Therefore, also in terms of privacy risk, the results of our embedding-based experiments match the pattern of the results of our rating-based methods.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and the associate editor for their valuable remarks and suggestions.

REFERENCES

- [1] Himan Abdollahpour, Masoud Mansouri, Robin Burke, and Bamshad Mobasher. 2019. The unfairness of popularity bias in recommendation. In *Proc. of the RMSE'19 Workshop, in Conjunction with ACM RecSys'19*.
- [2] Gediminas Adomavicius and Jingjing Zhang. 2012. Impact of data characteristics on recommender systems performance. *ACM Transactions on Management Information Systems* 3, 1 (2012), 1–17.
- [3] Sushant Agarwal. 2020. *Trade-offs between fairness, interpretability, and privacy in machine learning*. Master's thesis. University of Waterloo.
- [4] Vito Walter Anelli, Yashar Deldjoo, Tommaso Di Noia, Antonio Ferrara, and Fedelucio Narducci. 2021. How to put users in control of their data in federated top-N recommendation with learning to rank. In *Proc. of SAC'21*.
- [5] Ghazaleh Beigi and Huan Liu. 2020. A survey on privacy in social media: identification, mitigation, and applications. *ACM Transactions on Data Science* 1, 1 (2020), 1–38.
- [6] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. 2009. Pearson correlation coefficient. In *Noise Reduction in Speech Processing*. Springer, 37–40.
- [7] Shlomo Berkovsky, Tsvi Kuflik, and Francesco Ricci. 2012. The impact of data obfuscation on the accuracy of collaborative filtering. *Expert Systems with Applications* 39, 5 (2012), 5033–5042.
- [8] Michael Buckland and Fredric Gey. 1994. The relationship between recall and precision. *Journal of the American Society for Information Science* 45, 1 (1994), 12–19.
- [9] Joseph A. Calandrino, Ann Kilzer, Arvind Narayanan, Edward W. Felten, and Vitaly Shmatikov. 2011. “You might also like:” Privacy risks of collaborative filtering. In *Proc. of S&P'11*. IEEE, 231–246.
- [10] Di Chai, Leye Wang, Kai Chen, and Qiang Yang. 2022. Efficient federated matrix factorization against inference attacks. *ACM Transactions on Intelligent Systems and Technology* 13, 4, Article 59 (Jun 2022), 20 pages.
- [11] Chaochao Chen, Huiwen Wu, Jiajie Su, Lingjuan Lyu, Xiaolin Zheng, and Li Wang. 2022. Differential private knowledge transfer for privacy-preserving cross-domain recommendation. In *Proc. of ACM WWW'22*.
- [12] Chaochao Chen, Jun Zhou, Bingzhe Wu, Wenjing Fang, Li Wang, Yuan Qi, and Xiaolin Zheng. 2020. Practical privacy preserving POI recommendation. *ACM Transactions on Intelligent Systems and Technology* 11, 5, Article 52 (Jul 2020), 20 pages.
- [13] Xiaolin Chen, Xuemeng Song, Ruiyang Ren, Lei Zhu, Zhiyong Cheng, and Liqiang Nie. 2020. Fine-grained privacy detection with graph-regularized hierarchical attentive representation learning. *ACM Transactions on Information Systems* 38, 4 (2020), 1–26.
- [14] Ziqian Chen, Fei Sun, Yifan Tang, Haokun Chen, Jinyang Gao, and Bolin Ding. 2022. Studying the impact of data disclosure mechanism in recommender systems via simulation. *ACM Transactions on Information Systems* (2022).
- [15] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. 2021. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Transactions on Information Systems* 39, 2, Article 20 (Jan 2021), 49 pages.
- [16] Christian Desrosiers and George Karypis. 2010. A comprehensive survey of neighborhood-based recommendation methods. *Recommender Systems Handbook* (2010), 107–144.
- [17] Josep Domingo-Ferrer. 2010. Rational privacy disclosure in social networks. In *Proc. of MDAI'10*. Springer, 255–265.
- [18] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *Proc. of TAMC'08*. Springer, 1–19.
- [19] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proc. of ITCS'12*. 214–226.
- [20] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407.
- [21] Michael D. Ekstrand, Rezvan Josaghani, and Hoda Mehrpouyan. 2018. Privacy for all: Ensuring fair and equitable privacy protections. In *Proc. of FAccT'18*. PMLR, 35–47.
- [22] Jill Freyne and Shlomo Berkovsky. 2013. Evaluating recommender systems for supportive technologies. In *User Modeling and Adaptation for Daily Routines*. Springer, 195–217.
- [23] Arik Friedman, Bart P. Knijnenburg, Kris Vanhecke, Luc Martens, and Shlomo Berkovsky. 2015. Privacy aspects of recommender systems. In *Recommender Systems Handbook*. Springer, 649–688.
- [24] Chen Gao, Chao Huang, Dongsheng Lin, Depeng Jin, and Yong Li. 2020. DPLCF: Differentially private local collaborative filtering. In *Proc. of SIGIR'20*. 961–970.
- [25] Craig Gentry et al. 2009. *A Fully Homomorphic Encryption Scheme*. Stanford University, Stanford, CA.

- [26] Guibing Guo, Jie Zhang, Daniel Thalmann, and Neil Yorke-Smith. 2014. ETAF: An extended trust antecedents framework for trust prediction. In *Proc. of ASONAM'14*.
- [27] Jialiang Han, Yun Ma, Qiaozhu Mei, and Xuanzhe Liu. 2021. DeepRec: On-device deep learning for privacy-preserving sequential recommendation in mobile commerce. In *Proc. of WWW'21*. 900–911.
- [28] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems* 5, 4 (2015), 1–19.
- [29] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proc. of WWW'17*. 173–182.
- [30] Jonathan L. Herlocker, Joseph A. Konstan, A. I. Borchers, and John Riedl. 1999. An algorithmic framework for performing collaborative filtering. In *Proc. of SIGIR'99*. 230–237.
- [31] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* 22, 1 (Jan 2004), 5–53.
- [32] Dennis E. Hinkle, William Wiersma, and Stephen G. Jurs. 2003. *Applied Statistics for the Behavioral Sciences*. Vol. 663. Houghton Mifflin College Division.
- [33] T. Ryan Hoens, Marina Blanton, Aaron Steele, and Nitesh V. Chawla. 2013. Reliable medical recommendation systems with patient privacy. *ACM Transactions on Intelligent Systems and Technology* 4, 4, Article 67 (Oct 2013), 31 pages.
- [34] Longke Hu, Aixin Sun, and Yong Liu. 2014. Your neighbors affect your ratings: On geographical neighborhood influence to rating prediction. In *Proc. of SIGIR'14*.
- [35] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20, 4 (2002), 422–446.
- [36] Arjan J. P. Jeckmans, Michael Beye, Zekeriya Erkin, Pieter Hartel, Reginald L. Lagendijk, and Qiang Tang. 2013. Privacy in recommender systems. In *Social Media Retrieval*. Springer, 263–281.
- [37] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR'15*.
- [38] Bart P. Knijnenburg and Alfred Kobsa. 2013. Making decisions about privacy: Information disclosure in context-aware recommender systems. *ACM Transactions on Interactive Intelligent Systems* 3, 3 (2013), 1–23.
- [39] Dominik Kowald, Markus Schedl, and Elisabeth Lex. 2020. The unfairness of popularity bias in music recommendation: A reproducibility study. In *Proc. of ECIR'20*.
- [40] Emanuel Lacic, Dominik Kowald, Matthias Traub, Granit Luzhnica, Jörg Peter Simon, and Elisabeth Lex. 2015. Tackling cold-start users in recommender systems with indoor positioning systems. In *Proc. of ACM RecSys'15*. ACM.
- [41] Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Dongxiao Yu, Jun Ma, Maarten de Rijke, and Xiuzhen Cheng. 2020. Meta matrix factorization for federated rating predictions. In *Proc. of SIGIR'20*.
- [42] Kun Liu and Evmaria Terzi. 2010. A framework for computing the privacy scores of users in online social networks. *ACM Transactions on Knowledge Discovery from Data* 5, 1 (2010), 1–30.
- [43] Masoud Mansouri, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. 2020. Feedback loop and bias amplification in recommender systems. In *Proc. of CIKM'20*.
- [44] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Proc. of AISTATS'17*. PMLR, 1273–1282.
- [45] A. K. M. Nuhil Mehdy, Michael D. Ekstrand, Bart P. Knijnenburg, and Hoda Mehrpouyan. 2021. Privacy as a planned behavior: Effects of situational factors on privacy perceptions and plans. In *Proc. of UMAP'21*.
- [46] Peter Müllner, Dominik Kowald, and Elisabeth Lex. 2021. Robustness of meta matrix factorization against strict privacy constraints. In *Proc. of ECIR'21*.
- [47] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *Proc. of S&P'19*. IEEE.
- [48] Vasileios Perifanis and Pavlos S. Efraimidis. 2022. Federated neural collaborative filtering. *Knowledge-Based Systems* 242 (2022), 108441.
- [49] Naren Ramakrishnan, Benjamin J. Keller, Batul J. Mirza, Ananth Y. Grama, and George Karypis. 2001. When being weak is brave: Privacy in recommender systems. *IEEE Internet Computing* (2001), 54–62.
- [50] Hanchi Ren, Jingjing Deng, and Xianghua Xie. 2022. GRNN: Generative regression neural network—a data leakage attack for federated learning. *ACM Transactions on Intelligent Systems and Technology* 13, 4, Article 65 (May 2022), 24 pages.
- [51] Alan Said and Alejandro Bellogín. 2014. Comparative recommender system evaluation: benchmarking recommendation frameworks. In *Proc. of ACM RecSys'14*. 129–136.
- [52] Martin Saveski and Amin Mantrach. 2014. Item cold-start recommendations: Learning local collective embeddings. In *Proc. of ACM RecSys'14*. 89–96.
- [53] Agrima Srivastava and G. Geethakumari. 2013. Measuring privacy leaks in online social networks. In *Proc. of ICACCI'13*. IEEE, 2095–2100.

- [54] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from Transformer. In *Proc. of CIKM'19*. 1441–1450.
- [55] Qiang Tang and Jun Wang. 2016. Privacy-preserving friendship-based recommender systems. *IEEE Transactions on Dependable and Secure Computing* 15, 5 (2016), 784–796.
- [56] Isabel Wagner and David Eckhoff. 2018. Technical privacy metrics: A systematic survey. *ACM Computing Surveys (CSUR)* 51, 3 (2018), 1–38.
- [57] Mengting Wan and Julian J. McAuley. 2018. Item recommendation on monotonic behavior chains. In *Proc. of ACM RecSys'18*. 86–94.
- [58] Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian J. McAuley. 2019. Fine-grained spoiler detection from large-scale review corpora. In *Proc. of ACL'19*. Association for Computational Linguistics, 2605–2610.
- [59] Stanley L Warner. 1965. Randomized response: A survey technique for eliminating evasive answer bias. *J. Amer. Statist. Assoc.* 60, 309 (1965), 63–69.
- [60] Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Yongfeng Huang, and Xing Xie. 2022. FedCTR: Federated native Ad CTR prediction with cross-platform user behavior data. *ACM TIST* 13, 4, Article 62 (Jun 2022), 19 pages.
- [61] Yu Xin and Tommi Jaakkola. 2014. Controlling privacy in recommender systems. In *Proc. of NIPS'14*. 2618–2626.
- [62] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *Proc. of ICML'13*. PMLR, 325–333.
- [63] Mingwu Zhang, Yu Chen, and Jingqiang Lin. 2021. A privacy-preserving optimization of neighbourhood-based recommendation for medical-aided diagnosis and treatment. *IEEE Internet of Things Journal* (2021).
- [64] Minxing Zhang, Zhaochun Ren, Zihan Wang, Pengjie Ren, Zhunmin Chen, Pengfei Hu, and Yang Zhang. 2021. Membership inference attacks against recommender systems. In *Proc. of ACM SIGSAC'21*. 864–879.
- [65] Tianqing Zhu, Gang Li, Yongli Ren, Wanlei Zhou, and Ping Xiong. 2013. Differential privacy for neighborhood-based collaborative filtering. In *Proc. of IEEE/ACM ASONAM'13*. 752–759.

Received 17 August 2022; revised 26 May 2023; accepted 6 June 2023

P10 Differential Privacy in Collaborative Filtering Recommender Systems: A Review (2023)

Privacy and Limited Preference Information in Recommender Systems

[P10] Muellner P., Lex, E., Schedl, M., **Kowald, D.** (2023). Differential Privacy in Collaborative Filtering Recommender Systems: A Review. *Frontiers in Big Data*, 6:1249997, pp. 1-7. DOI: <https://doi.org/10.3389/fdata.2023.1249997>



OPEN ACCESS

EDITED BY

Yassine Himeur,
University of Dubai, United Arab Emirates

REVIEWED BY

Chaochao Chen,
Zhejiang University, China

*CORRESPONDENCE

Peter Müllner
✉ pmuellner@know-center.at;
✉ pmuellner@student.tugraz.at
Elisabeth Lex
✉ elisabeth.lex@tugraz.at
Dominik Kowald
✉ dkowald@know-center.at

RECEIVED 29 June 2023

ACCEPTED 25 September 2023

PUBLISHED 12 October 2023

CITATION

Müllner P, Lex E, Schedl M and Kowald D (2023) Differential privacy in collaborative filtering recommender systems: a review. *Front. Big Data* 6:1249997. doi: 10.3389/fdata.2023.1249997

COPYRIGHT

© 2023 Müllner, Lex, Schedl and Kowald. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](#). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Differential privacy in collaborative filtering recommender systems: a review

Peter Müllner^{1,2*}, Elisabeth Lex^{2*}, Markus Schedl^{3,4} and Dominik Kowald^{1,2*}

¹Know-Center Gmbh, Graz, Austria, ²Institute of Interactive Systems and Data Science, Graz University of Technology, Graz, Austria, ³Institute of Computational Perception, Johannes Kepler University Linz, Linz, Austria, ⁴Linz Institute of Technology, Linz, Austria

State-of-the-art recommender systems produce high-quality recommendations to support users in finding relevant content. However, through the utilization of users' data for generating recommendations, recommender systems threaten users' privacy. To alleviate this threat, often, differential privacy is used to protect users' data via adding random noise. This, however, leads to a substantial drop in recommendation quality. Therefore, several approaches aim to improve this trade-off between accuracy and user privacy. In this work, we first overview threats to user privacy in recommender systems, followed by a brief introduction to the differential privacy framework that can protect users' privacy. Subsequently, we review recommendation approaches that apply differential privacy, and we highlight research that improves the trade-off between recommendation quality and user privacy. Finally, we discuss open issues, e.g., considering the relation between privacy and fairness, and the users' different needs for privacy. With this review, we hope to provide other researchers an overview of the ways in which differential privacy has been applied to state-of-the-art collaborative filtering recommender systems.

KEYWORDS

differential privacy, collaborative filtering, recommender systems, accuracy-privacy trade-off, review

1. Introduction

Several previous research works have revealed multiple privacy threats for users in recommender systems. For example, the disclosure of users' private data to untrusted third parties ([Calandriano et al., 2011](#)), or the inference of users' sensitive attributes, such as gender or age ([Zhang et al., 2023](#)). Similarly, also the users themselves care more about their privacy in recommender systems ([Herbert et al., 2021](#)). For these reasons, privacy-enhancing techniques have been applied, most prominently *differential privacy* (DP) ([Dwork, 2008](#)). DP injects random noise into the recommender system and formally guarantees a certain degree of privacy. However, through this random noise, the quality of the recommendations suffers ([Berkovsky et al., 2012](#)). Many works aim to address this trade-off between recommendation quality and user privacy via carefully applying DP in specific ways. [Friedman et al. \(2016\)](#) show that in case of matrix factorization, DP can be applied to three different parts of the recommender system: (i) to the input of the recommender system, (ii) within the training process of the model, and (iii) to the model after training. However, a concise overview of works with respect to these three categories does not exist yet.

Therefore, in the paper at hand, we address this gap and identify 26 papers from relevant venues that deal with DP in collaborative filtering recommender systems. We briefly review these 26 papers and make two key observations about the state-of-the-art. Firstly, the vast majority of works use datasets from the same non-sensitive domain, i.e., movies. Secondly, research on applying DP after model training is scarce. Finally, we discuss our findings and present two open questions that may be relevant for future research: *How does applying DP impact fairness?* and *How to quantify the user's perceived privacy?*

Our work is structured as follows: In Section 2, we present threats to the privacy of users in recommender systems and additionally, introduce the DP framework. In Section 3, we precisely outline our methodology for obtaining the set of 26 relevant papers. In Section 4, we review these papers and group them into three groups according to the way in which they apply DP. In Section 5, we discuss our findings and propose open issues that we identified.

2. Background

In recent years, users of recommender systems have shown increasing concerns with respect to keeping their data private (Herbert et al., 2021). In fact, several research works (Bilge et al., 2013; Jeckmans et al., 2013; Friedman et al., 2015; Beigi and Liu, 2020; Majeed and Lee, 2020; Himeur et al., 2022) have revealed multiple privacy threats, for example, the inadvertent disclosure of users' interaction data, or the inference of users' sensitive attributes (e.g., gender, age).

Typically, a recommender system utilizes historic interaction data to generate recommendations. Ramakrishnan et al. (2001) show that in k nearest neighbors recommender systems, the recommendations could disclose the interaction data of the neighbors, i.e., users, whose interaction data is utilized to generate the recommendations. Similarly, Calandrino et al. (2011) inject fake users to make the recommendations more likely to disclose the neighbors' interaction data, and also, they can infer users' interaction data based on the public outputs of a recommender system, e.g., public interaction data or public product reviews. Furthermore, Hashemi et al. (2022) and Xin et al. (2023) aim to learn user behavior via observing many recommendations and, in this way, can disclose parts of a user's interaction data. Weinsberg et al. (2012) show that an adversary could infer sensitive attributes, in this case, gender, based on a user's interaction data. Their attack relies on a classifier that leverages a small set of training examples to learn the correlation between a user's preferences and gender. Likewise, Ganhör et al. (2022) show that recommender systems based on autoencoder architectures are vulnerable to infer the user's gender from the latent user representation. The authors also propose an adversarial training regime to mitigate this problem. Similarly, also Zhang et al. (2023) infer the age and gender of users in a federated learning recommender system. In summary, many of a user's sensitive attributes can be inferred via thoroughly analyzing the user's digital footprint (e.g., the behavior in a recommender system or social media platform) (Kosinski et al., 2013).

Overall, the utilization of users' interaction data for generating recommendations poses a privacy risk for users. Therefore, privacy-enhancing techniques, such as homomorphic encryption (Gentry, 2009), federated learning (McMahan et al., 2017), or most prominently, *differential privacy* (DP) (Dwork, 2008) have been applied to protect users' privacy. Specifically, DP is applied via injecting noise into the recommender system. This ensures that the recommender system uses noisy data instead of the real data. For example, an additive mechanism samples random noise from the Laplace or Gaussian distribution and adds it to the users' rating data (Dwork and Roth, 2014). Alternatively, the randomized responses mechanism flips a fair coin, which decides whether to use the real data or random data, and this way, ensures DP (Warner, 1965; Dwork and Roth, 2014). Overall, the degree of noise that is used is defined by the parameter ϵ , i.e., the privacy budget. Intuitively, the smaller the ϵ -value is, the better the privacy, but the stronger the expected accuracy drop. Therefore, choosing ϵ is non-trivial and depends on the specific use case (Dwork, 2008).

3. Review methodology

To conduct our review, we chose relevant conferences in the field, i.e., ACM SIGIR, TheWebConf, ACM KDD, IJCAI, ACM CIKM, and ACM RecSys and journals, i.e., TOIS, TIST, UMUAI, and TKDE. Adopting a keyword-based search, we identified relevant publications in the proceedings via querying the full-texts for "differential privacy" and "recommender system", "recommend", "recommendation", or "recommender". We manually checked the resulting papers for their relevance and retrieved 16 publications. In addition, we conducted a literature search on Google Scholar using the same keywords and procedure, which resulted in 10 publications. Overall, we considered 26 publications in the paper at hand.

4. Recommender systems with differential privacy

According to Friedman et al. (2016), DP can be applied via (i) adding noise to the input of a collaborative filtering-based recommender system, e.g., the user data or other user representations, (ii) adding noise to the training process of the model, i.e., the model updates, or (iii) adding noise to the model after training, i.e., to the resulting latent factors. In Table 1, we group the selected publications into these three categories.

4.1. Differential privacy applied to the user representation

In collaborative filtering recommender systems, the input to the system is typically given by interaction or rating data. However, more complex user representations exist, e.g., neural-based user embeddings.

Chen et al. (2020) protect POI (point of interest) interaction data of users, e.g., a user visited a restaurant, with DP. Specifically, they use this data to privately calculate POI features, i.e., the

TABLE 1 Overview of the reviewed 26 publications.

References	Domain(s)	DP applied to		
		User represent.	Model updates	After training
Long et al. (2023)	Location	•		
Müllner et al. (2023)	Movies, Music, Books, Social	•		
Neera et al. (2023)	Movies, Jokes, Dating	•		
Wang et al. (2023)	Movies, Music		•	
Chai et al. (2022)	Movies, Location	•		
Chen et al. (2022)	Movies, Music, Books	•		
Jiang et al. (2022)	Movies, Music, Location, Groceries		•	
Liu et al. (2022)	Social		•	
Ning et al. (2022)	Movies		•	
Ran et al. (2022)	Movies, Music			•
Ren et al. (2022)	Social	•		
Wu et al. (2022)	Advertisement	•		
Li et al. (2021)	Movies, Dating		•	
Minto et al. (2021)	Movies		•	
Zhang et al. (2021)	Movies	•		•
Chen et al. (2020)	Location	•		
Gao et al. (2020)	Movies, Smartphone	•		
Ma et al. (2019)	Health		•	
Meng et al. (2018)	Social		•	
Shin et al. (2018)	Movies, Dating		•	
Liu et al. (2017)	Movies	•		
Yang et al. (2017)	Movies	•		
Li et al. (2016)	Movies	•		
Hua et al. (2015)	Movies		•	•
Zhu et al. (2013)	Movies	•		
Zhao et al. (2011)	Movies	•		

We mark whether DP is applied to the user representation, to the model updates, or after training. Domain(s) refers to the domain(s) in which the recommendations are evaluated. We sort the publications with respect to recency.

number of visitors per restaurant, which are subsequently used for generating recommendations instead of the DP-protected interaction data. This way, they can increase recommendation accuracy. Similarly, Long et al. (2023) use DP to recommend POIs, but in a decentralized fashion. A central server collects public data to train a recommendation model and to privately identify groups of similar users. DP is used for privately calculating user-user similarities. Then, users locally use information from similar users, which leads to a better trade-off between recommendation quality and privacy than comparable approaches.

Liu et al. (2017) add noise to users' rating data and to the user-user covariance matrix to ensure DP of a KNN-based recommender system. They show that this leads to better privacy than in case only the covariance matrix is protected via DP. Besides revealing users' rating data, an attacker could also aim to infer sensitive attributes (e.g., gender) of the users. Therefore, Chai et al. (2022) propose an obfuscation model to protect gender information. After applying

this obfuscation model, users protect their data via DP and send it to a central server. Yang et al. (2017) use the Johnson-Lindenstrauss transform (Blocki et al., 2012), i.e., they ensure DP via multiplying the original interaction matrix with a random matrix. Using this protected matrix, their approach guarantees differential privacy and also can even generate more accurate recommendations than a non-private approach. Neera et al. (2023) underline that adding Laplacian noise can lead to "unrealistic" rating values, i.e., outside the rating range, and through this, recommendation accuracy can drop severely. Therefore, they bound the noisy ratings to a "realistic" value range without harming DP. Plus, they use a Gaussian mixture model to estimate and then remove noise in the recommendation process to keep recommendation accuracy.

Cross-domain recommendation models can increase recommendation accuracy in the target domain by exploiting data from multiple source domains. To protect user privacy when data from the source domain is made available to the target domain,

Chen et al. (2022) use the Johnson-Lindenstrauss transform. Due to the high sparsity of the rating matrix, they employ a variant that performs better when applied to sparse matrices (Ailon and Chazelle, 2009). Ren et al. (2022) utilize data from different social network platforms to generate recommendations and apply DP to the user attributes and the connections in the social network graphs. Plus, they apply a variant of DP to protect textual data (Fernandes et al., 2019). Moreover, to increase the click-through rate for recommended advertisements, Wu et al. (2022) leverage user interaction data from multiple platforms. First, user embeddings are generated per platform and then protected with DP. Second, the recommender system collects and aggregates a user's DP-protected embeddings across platforms and then applies DP again to the aggregated user embedding. According to the authors, applying DP after aggregation allows for smaller noise levels when applying DP to the per-platform user embeddings, which results in higher accuracy. Typically, many users use a variety of different online platforms. Therefore, Li et al. (2016) leverage these multiple data sources per user to increase recommendation accuracy. Specifically, they combine DP-protected item-item similarities from dataset *B* as auxiliary data that helps to generate more accurate recommendations for users in dataset *A* (cf. Zhao et al., 2011).

Gao et al. (2020) compute item-item similarities by using DP-protected user interaction data. With these item-item similarities, users can locally generate recommendations on their own devices, therefore not harming their privacy. The item-based KNN recommender system proposed by Zhu et al. (2013) utilizes DP in two ways: First, they randomly rearrange the most similar neighbors to foster privacy. Second, they measure how the item-item similarity changes if a specific user interaction was not present, and with this, they add the necessary level of noise to the users' interactions. This way, recommendation accuracy can be better preserved than with approaches that apply the same level of noise to all user interactions. For user-based KNN, Müllner et al. (2023) identify neighbors that can be reused for many recommendations. This way, only a small set of users are used as neighbors for many recommendations and need to be protected with DP. Many users, however, are only rarely utilized as neighbors and therefore do not need to be protected with DP. Overall, this yields more accurate recommendations than in case DP needs to be applied to all users.

4.2. Differential privacy applied to the model updates

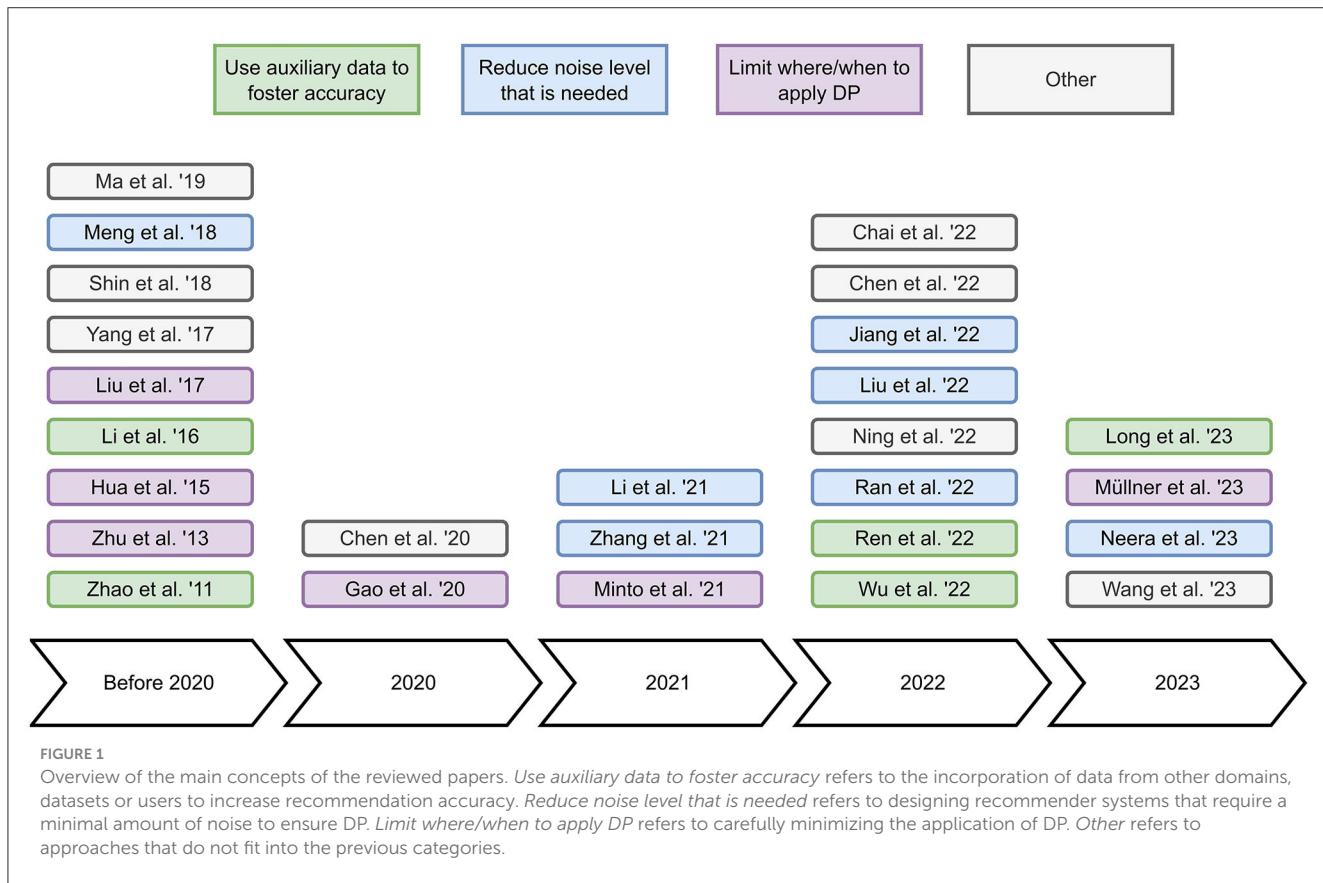
Some recommender systems do not process user data and create user representations on a central server, instead, they compute the model updates, i.e., gradients, locally on their users' device. Then, the recommender system collects these gradients to adapt its recommendation model. To prohibit the leakage of user data through these gradients (Bagdasaryan et al., 2020), DP can be applied.

For example, Hua et al. (2015) add noise to the gradients of the recommendation model to ensure DP. However, due to the sparsity of the gradients, the application of DP can be ineffective and information about what items have been rated by the user

can be disclosed. To address this problem, Shin et al. (2018) use DP to mask whether a user appears in the dataset. Also, they formally show that the noise added to the gradients hinders a fast convergence of the recommendation model, and in this way, increases the training time. Therefore, they introduce a stabilization factor to enable better training of the recommendation model. Wang et al. (2023) propose a recommender system that uses a special DP-mechanism (Zhao et al., 2020) to simultaneously protect the rating values and the set of items that is rated by a user. The DP-protected item-vectors are then send to a central server, which performs dimensionality reduction to reduce the accuracy drop (cf. Shin et al., 2018). In Minto et al. (2021), users receive a global model from a central server and, then, compute their respective updates locally. These updates are protected via DP, before being sent back to the server. Plus, the number of updates per user are restricted to further improve privacy. Moreover, the authors highlight that high-dimensional gradients can negatively impact the recommendation quality, as they are especially prone to higher sparsity (cf. Hua et al., 2015; Shin et al., 2018). When DP is applied, the gradients become denser since noise is added to the entire gradient, including the zero-entries. This, in practice, leads to additional communication overhead, since all non-zero-entries need to be transmitted (Ning et al., 2022). Therefore, Ning et al. only add noise to the non-zero gradients. This way, the communication overhead is reduced; however, DP cannot be guaranteed anymore.

Jiang et al. (2022) reduce the accuracy drop via an adaptive DP mechanism that depends on the number of training steps. Intuitively, after many training steps, the model fine-tunes its predictions and the gradients need to be measured more accurately than during the beginning of the model training. Thus, they add more noise in the beginning and less noise in the end of the training process. This yields more accurate recommendations than a static DP mechanism that always adds the same level of noise. Li et al. (2021) also use noisy model updates to ensure DP. They observe that noise can lead to large values for the user embeddings, which increases the sensitivity and therefore also the level of noise that is required to ensure DP. To foster recommendation quality, they map the user embeddings to a certain range, which bounds the sensitivity and requires less noise. Liu et al. (2022) leverage user interactions and social connections to generate recommendations via a federated graph neural network. To ensure DP, they add noise to the gradients that are sent to a central server. However, gradients with different magnitudes have different sensitivities (cf. Li et al., 2021), and thus, need a different level of noise to ensure DP. Therefore, they fit the noise level to the gradient magnitudes to satisfy DP, but also, to preserve recommendation accuracy.

Ma et al. (2019) employ federated tensor factorization in the health domain. A global model is distributed to hospitals, which locally update the model based on their data. To protect privacy, a variant of DP is applied to the model updates, which are subsequently sent to the global server to adapt the global model. Meng et al. (2018) randomly divide users' ratings into non-sensitive and sensitive ratings. For sensitive ratings, they apply more noise than for non-sensitive ratings. With this, their approach can preserve higher recommendation accuracy than in case the same noise level is used for sensitive and non-sensitive data.



4.3. Differential privacy applied after training

Only few works apply DP to the recommendation model after training. In case of a matrix factorization approach, noise can be added to the learned user- and item-vectors to ensure DP. Our selected publications (see Section 3) do not include any works that apply DP exclusively to the model after training. Nevertheless, we describe works that apply DP to the user representation or the model updates, but also after training.

For example, Hua et al. (2015) consider a matrix factorization model, where the model sends item-vectors back to the users and this way, users' data can get leaked. To prohibit this, Hua et al. perturb the model's objective function after training via adding noise to the latent item-vectors. Similarly, Ran et al. (2022) also use DP to prohibit data leakage through the item-vectors that are sent to the users. Specifically, a trusted recommender system generates a matrix factorization model. Instead of publishing the item-vectors of this model, they learn new item-vectors on the DP-protected user-vectors. Through this, they can minimize the noise that is introduced and thus, can improve recommendation accuracy over comparable approaches. Zhang et al. (2021) apply DP to the user representation and also, to the model after training. Specifically, they use a polynomial approximation of the model's loss function to efficiently compute the sensitivity of the dataset and, accordingly, adapt the level of noise that is added to the loss function.

5. Summary and open questions

In this review, we investigate research works that apply DP to collaborative filtering recommender systems. We identify 26 relevant works and categorize these based on how they apply DP, i.e., to the user representation, to the model updates, or to the model after training (see Table 1). In addition, we briefly summarize these relevant works to obtain a broad overview of the state-of-the-art. Furthermore, we identify the main concepts of the relevant works in Figure 1 to help readers to understand in which diverse ways the reviewed papers apply DP to improve the accuracy-privacy trade-off. Our main findings from reviewing the discussed literature are two-fold: (i) The majority of works use datasets from the same non-sensitive domain, i.e., movies, and (ii) applying DP to the model after training seems to be an understudied topic.

Many research works use datasets from the movie domain, which, in general, does not include sensitive data. For research on DP in collaborative filtering recommender systems, however, datasets from sensitive domains may be better suited to resemble real-world privacy threats well. For example, datasets from the health, finance, or job domain. Moreover, the majority of research focuses on either applying DP to the user representation or to the model updates. Research on applying DP to the model after training is scarce, and therefore, this opens up the possibility of future work to fill this gap.

Our review of relevant work allows to grasp the state-of-the-art and to identify the following open research questions:

Q1: How does applying DP impact fairness? Dwork et al. (2012) and Zemel et al. (2013) suggest that in theory, privacy can lead to fairness and fairness can lead to privacy. The reason is that for both, a user's data shall be hidden, either to ensure privacy or to prohibit discrimination based on this data. However, in practice, correlations in private data can still lead to unfairness (Ekstrand et al., 2018; Agarwal, 2020). Only recently, Yang et al. (2023) and Sun et al. (2023) investigate the connection between privacy and fairness in recommender systems. For example, Sun et al. (2023) use DP-protected information to re-rank the items in the recommendation list and in this way, increase a more fair exposure of items. Nonetheless, the impact of DP on fairness remains an understudied topic.

Q2: How to quantify the user's perceived privacy? Users perceive privacy differently, e.g., some users tolerate disclosing their gender, while others refuse to do this (Joshaghani et al., 2018). This perceived privacy depends on many factors, e.g., context or situational factors (Knijnenburg and Kobsa, 2013; Mehdy et al., 2021). However, measuring users' perceived privacy is hard and is usually done via questionnaires (Knijnenburg and Kobsa, 2013). This is in stark contrast to how privacy is measured in the DP framework, i.e., via quantifying to what extent the data impacts the output of the recommender system. Therefore, developing methods to better quantify users' privacy is an important future research avenue.

Author contributions

PM: literature analysis, conceptualization, and writing. MS: conceptualization and writing. EL and DK: conceptualization, writing, and supervision. All authors contributed to the article and approved the submitted version.

References

- Agarwal, S. (2020). *Trade-offs between fairness, interpretability, and privacy in machine learning* (Master's thesis). University of Waterloo, Waterloo, ON, Canada.
- Ailon, N., and Chazelle, B. (2009). The fast Johnson–Lindenstrauss transform and approximate nearest neighbors. *SIAM J. Comput.* 39, 302–322. doi: 10.1137/060673096
- Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., and Shmatikov, V. (2020). “How to backdoor federated learning,” in *International Conference on Artificial Intelligence and Statistics* (Palermo: PMLR), 2938–2948.
- Beigi, G., and Liu, H. (2020). A survey on privacy in social media: identification, mitigation, and applications. *ACM Trans. Data Sci.* 1, 1–38. doi: 10.1145/3343038
- Berkovsky, S., Kuflik, T., and Ricci, F. (2012). The impact of data obfuscation on the accuracy of collaborative filtering. *Expert Syst. Appl.* 39, 5033–5042. doi: 10.1016/j.eswa.2011.11.037
- Bilge, A., Kaleli, C., Yakut, I., Gunes, I., and Polat, H. (2013). A survey of privacy-preserving collaborative filtering schemes. *Int. J. Softw. Eng. Knowledge Eng.* 23, 1085–1108. doi: 10.1142/S0218194013500320
- Blocki, J., Blum, A., Datta, A., and Sheffet, O. (2012). “The Johnson–Lindenstrauss transform itself preserves differential privacy,” in *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science* (New Brunswick, NJ: IEEE), 410–419.
- Calandrino, J. A., Kilzer, A., Narayanan, A., Felten, E. W., and Shmatikov, V. (2011). “You might also like:” privacy risks of collaborative filtering,” in *Proc. of Se-P’11* (Oakland, CA: IEEE), 231–246.
- Chai, D., Wang, L., Chen, K., and Yang, Q. (2022). Efficient federated matrix factorization against inference attacks. *ACM Trans. Intell. Syst. Technol.* 13, 1–20. doi: 10.1145/3501812
- Chen, C., Wu, H., Su, J., Lyu, L., Zheng, X., and Wang, L. (2022). “Differential private knowledge transfer for privacy-preserving cross-domain recommendation,” in *Proc. of ACM WWW’22* (Lyon).
- Chen, C., Zhou, J., Wu, B., Fang, W., Wang, L., Qi, Y., et al. (2020). Practical privacy preserving POI recommendation. *ACM Trans. Intell. Syst. Technol.* 11, 1455–1465. doi: 10.1145/3394138
- Dwork, C. (2008). “Differential privacy: a survey of results,” in *International Conference on Theory and Applications of Models of Computation* (Berlin: Springer), 1–19.
- Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). “Fairness through awareness,” in *Proc. of ITCS’12* (Cambridge, MA), 214–226.
- Dwork, C., and Roth, A. (2014). The algorithmic foundations of differential privacy. *Found. Trends Theoret. Comput. Sci.* 9, 211–407. doi: 10.1561/0400000042
- Ekstrand, M. D., Joshaghani, R., and Mehrpouyan, H. (2018). “Privacy for all: Ensuring fair and equitable privacy protections,” in *Proc. of FAccT’18* (New York, NY: PMLR), 35–47.
- Fernandes, N., Dras, M., and McIver, A. (2019). “Generalised differential privacy for text document processing,” in *Principles of Security and Trust: 8th International Conference, POST 2019* (Prague: Springer International Publishing), 123–148.
- Friedman, A., Berkovsky, S., and Kaafar, M. A. (2016). A differential privacy framework for matrix factorization recommender systems. *User Model. User Adapt. Interact.* 26, 425–458. doi: 10.1007/s11257-016-9177-7
- Friedman, A., Knijnenburg, B. P., Vanhecke, K., Martens, L., and Berkovsky, S. (2015). “Privacy aspects of recommender systems,” in *Recommender Systems Handbook*, eds F. Ricci, L. Rokach, and B. Shapira (Boston, MA: Springer), 649–688.

Funding

This work was supported by the DDAI COMET Module within the COMET-Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT), the Austrian Federal Ministry for Digital and Economic Affairs (BMDW), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG), and partners from industry and academia. The COMET Programme is managed by FFG. In addition, the work received funding from the TU Graz Open Access Publishing Fund and from the Austrian Science Fund (FWF): DFH-23 and P33526.

Conflict of interest

PM was employed by Know-Center GmbH.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Ganhör, C., Penz, D., Rekabsaz, N., Lesota, O., and Schedl, M. (2022). “Unlearning protected user attributes in recommendations with adversarial training,” in *SIGIR ’22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, eds E. Amigó, P. Castells, J. Gonzalo, B. Carterette, J. S. Culpepper, and G. Kazai (Madrid: ACM), 2142–2147.
- Gao, C., Huang, C., Lin, D., Jin, D., and Li, Y. (2020). “DPLCF: differentially private local collaborative filtering,” in *Proc. of SIGIR’20* (Xi’an), 961–970.
- Gentry, C. (2009). *A Fully Homomorphic Encryption Scheme* (Dissertation). Stanford University.
- Hashemi, H., Xiong, W., Ke, L., Maeng, K., Annavarapu, M., Suh, G. E., et al. (2022). Data leakage via access patterns of sparse features in deep learning-based recommendation systems. *arXiv preprint arXiv:2212.06264*.
- Herbert, C., Marschin, V., Erb, B., Meißner, D., Aufheimer, M., and Bösch, C. (2021). Are you willing to self-disclose for science? Effects of privacy awareness and trust in privacy on self-disclosure of personal and health data in online scientific studies—an experimental study. *Front. Big Data* 4, 763196. doi: 10.3389/fdata.2021.763196
- Himeur, Y., Sohail, S. S., Bensaali, F., Amira, A., and Alazab, M. (2022). Latest trends of security and privacy in recommender systems: a comprehensive review and future perspectives. *Comput. Sec.* 118, 102746. doi: 10.1016/j.cose.2022.102746
- Hua, J., Xia, C., and Zhong, S. (2015). “Differentially private matrix factorization,” in *International Joint Conference on Artificial Intelligence* (Buenos Aires).
- Jeckmans, A. J., Beye, M., Erkin, Z., Hartel, P., Lagendijk, R. L., and Tang, Q. (2013). “Privacy in recommender systems,” in *Social Media Retrieval*, eds N. Ramzan, R. van Zwol, J.-S. Lee, K. Clüver, and X.-S. Hua (London: Springer), 263–281.
- Jiang, X., Liu, B., Qin, J., Zhang, Y., and Qian, J. (2022). “FedNCF: federated neural collaborative filtering for privacy-preserving recommender system,” in *2022 International Joint Conference on Neural Networks (IJCNN)* (Padua: IEEE), 1–8.
- Joshaghani, R., Ekstrand, M. D., Knijnenburg, B., and Mehrpouyan, H. (2018). “Do different groups have comparable privacy tradeoffs?” in *Workshop on Moving Beyond a “One-Size Fits All” Approach: Exploring Individual Differences in Privacy, in Conjunction with the ACM CHI Conference on Human Factors in Computing Systems, CHI 2018* (Montreal, QC).
- Knijnenburg, B. P., and Kobsa, A. (2013). Making decisions about privacy: information disclosure in context-aware recommender systems. *ACM Trans. Interact. Intell. Syst.* 3, 1–23. doi: 10.1145/2499670
- Kosinski, M., Stillwell, D., and Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *Proc. Nat. Acad. Sci. U.S.A.* 110, 5802–5805. doi: 10.1073/pnas.1218772110
- Li, J., Yang, J.-J., Zhao, Y., Liu, B., Zhou, M., Bi, J., and Wang, Q. (2016). Enforcing differential privacy for shared collaborative filtering. *IEEE Access* 5, 35–49. doi: 10.1109/ACCESS.2016.2600258
- Li, Z., Ding, B., Zhang, C., Li, N., and Zhou, J. (2021). Federated matrix factorization with privacy guarantee. *Proc. VLDB Endowment* 15, 900–913. doi: 10.14778/3503585.3503598
- Liu, X., Liu, A., Zhang, X., Li, Z., Liu, G., Zhao, L., and Zhou, X. (2017). “When differential privacy meets randomized perturbation: a hybrid approach for privacy-preserving recommender system,” in *International Conference on Database Systems for Advanced Applications* (Suzhou: Springer), 576–591.
- Liu, Z., Yang, L., Fan, Z., Peng, H., and Yu, P. S. (2022). Federated social recommendation with graph neural network. *ACM Trans. Intell. Syst. Technol.* 13, 1–24. doi: 10.1145/3501815
- Long, J., Chen, T., Nguyen, Q. V. H., and Yin, H. (2023). Decentralized collaborative learning framework for next POI recommendation. *ACM Trans. Inf. Syst.* 41, 1–25. doi: 10.1145/3555374
- Ma, J., Zhang, Q., Lou, J., Ho, J. C., Xiong, L., and Jiang, X. (2019). “Privacy-preserving tensor factorization for collaborative health data analysis,” in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM ’19* (New York, NY: Association for Computing Machinery), 1291–1300.
- Majeed, A., and Lee, S. (2020). Anonymization techniques for privacy preserving data publishing: a comprehensive survey. *IEEE Access* 9, 8512–8545. doi: 10.1109/ACCESS.2020.3045700
- McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. (2017). “Communication-efficient learning of deep networks from decentralized data,” in *Artificial Intelligence and Statistics* (Fort Lauderdale, FL: PMLR), 1273–1282.
- Mehdy, A. N., Ekstrand, M. D., Knijnenburg, B. P., and Mehrpouyan, H. (2021). “Privacy as a planned behavior: effects of situational factors on privacy perceptions and plans,” in *Proc. of UMAP’21* (Utrecht).
- Meng, X., Wang, S., Shu, K., Li, J., Chen, B., Liu, H., et al. (2018). “Personalized privacy-preserving social recommendation,” in *Proceedings of the AAAI Conference on Artificial Intelligence* (New Orleans, LA).
- Minto, L., Haller, M., Livshits, B., and Haddadi, H. (2021). “Stronger privacy for federated collaborative filtering with implicit feedback,” in *Proceedings of the 15th ACM Conference on Recommender Systems* (Amsterdam), 342–350.
- Müllner, P., Lex, E., Schedl, M., and Kowald, D. (2023). ReuseKNN: neighborhood reuse for differentially-private KNN-based recommendations. *ACM Trans. Intell. Syst. Technol.* 14, 1–29. doi: 10.1145/3608481
- Neera, J., Chen, X., Aslam, N., Wang, K., and Shu, Z. (2023). Private and utility enhanced recommendations with local differential privacy and Gaussian mixture model. *IEEE Trans. Knowledge Data Eng.* 35, 4151–4163. doi: 10.1109/TKDE.2021.3126577
- Ning, L., Chien, S., Song, S., Chen, M., Xue, Y., and Berlowitz, D. (2022). “EANA: reducing privacy risk on large-scale recommendation models,” in *Proceedings of the 16th ACM Conference on Recommender Systems* (Seattle, WA), 399–407.
- Ramakrishnan, N., Keller, B. J., Mirza, B. J., Grama, A. Y., and Karaypis, G. (2001). When being weak is brave: privacy in recommender systems. *IEEE Internet Comput.* 5, 54–62. doi: 10.1109/4236.968832
- Ran, X., Wang, Y., Zhang, L. Y., and Ma, J. (2022). A differentially private matrix factorization based on vector perturbation for recommender system. *Neurocomputing* 483, 32–41. doi: 10.1016/j.neucom.2022.01.079
- Ren, J., Jiang, L., Peng, H., Lyu, L., Liu, Z., Chen, C., et al. (2022). “Cross-network social user embedding with hybrid differential privacy guarantees,” in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM ’22* (New York, NY: Association for Computing Machinery), 1685–1695.
- Shin, H., Kim, S., Shin, J., and Xiao, X. (2018). Privacy enhanced matrix factorization for recommendation with local differential privacy. *IEEE Trans. Knowledge Data Eng.* 30, 1770–1782. doi: 10.1109/TKDE.2018.2805356
- Sun, J. A., Pentyala, S., Cock, M. D., and Farnadi, G. (2023). “Privacy-preserving fair item ranking,” in *Advances in Information Retrieval*, eds J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, and A. Caputo (Cham: Springer Nature), 188–203.
- Wang, Y., Gao, M., Ran, X., Ma, J., and Zhang, L. Y. (2023). An improved matrix factorization with local differential privacy based on piecewise mechanism for recommendation systems. *Expert Syst. Appl.* 216, 119457. doi: 10.1016/j.eswa.2022.119457
- Warner, S. L. (1965). Randomized response: a survey technique for eliminating evasive answer bias. *J. Am. Stat. Assoc.* 60, 63–69.
- Weinsberg, U., Bhagat, S., Ioannidis, S., and Taft, N. (2012). “Blurme: inferring and obfuscating user gender based on ratings,” in *Proceedings of the Sixth ACM Conference on Recommender Systems* (Dublin), 195–202.
- Wu, C., Wu, F., Lyu, L., Huang, Y., and Xie, X. (2022). FedCTR: Federated native ad CTR prediction with cross-platform user behavior data. *ACM TIST* 13, 1–19. doi: 10.1145/3506715
- Xin, X., Yang, J., Wang, H., Ma, J., Ren, P., Luo, H., et al. (2023). On the user behavior leakage from recommender system exposure. *ACM Trans. Inform. Syst.* 41, 1–25. doi: 10.1145/3568954
- Yang, M., Zhu, T., Ma, L., Xiang, Y., and Zhou, W. (2017). “Privacy preserving collaborative filtering via the Johnson-Lindenstrauss transform,” in *2017 IEEE Trustcom/BigDataSE/ICESS* (Sydney, NSW: IEEE), 417–424.
- Yang, Z., Ge, Y., Su, C., Wang, D., Zhao, X., and Ying, Y. (2023). “Fairness-aware differentially private collaborative filtering,” in *Companion Proceedings of the ACM Web Conference 2023, WWW ’23 Companion* (Austin, TX: Association for Computing Machinery), 927–931.
- Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. (2013). “Learning fair representations,” in *Proc. of ICML’13* (Atlanta, GA: PMLR), 325–333.
- Zhang, S., Yin, H., Chen, T., Huang, Z., Cui, L., and Zhang, X. (2021). “Graph embedding for recommendation against attribute inference attacks,” in *Proceedings of the Web Conference 2021, WWW ’21* (New York, NY: Association for Computing Machinery), 3002–3014.
- Zhang, S., Yuan, W., and Yin, H. (2023). Comprehensive privacy analysis on federated recommender system against attribute inference attacks. *IEEE Trans. Knowledge Data Eng.* 1–13. doi: 10.1109/TKDE.2023.3295601
- Zhao, Y., Feng, X., Li, J., and Liu, B. (2011). “Shared collaborative filtering,” in *Proceedings of the Fifth ACM Conference on Recommender Systems* (Chicago, IL), 29–36.
- Zhao, Y., Zhao, J., Yang, M., Wang, T., Wang, N., Lyu, L., et al. (2020). Local differential privacy-based federated learning for internet of things. *IEEE Internet Things J.* 8, 8836–8853. doi: 10.1109/JIOT.2020.3037194
- Zhu, T., Li, G., Ren, Y., Zhou, W., and Xiong, P. (2013). “Differential privacy for neighborhood-based collaborative filtering,” in *Proc. of IEEE/ACM ASONAM’13* (Niagara Falls, ON), 752–759.

B.3 Fairness and Popularity Bias in Recommender Systems

P11 The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study (2020)

Fairness and Popularity Bias in Recommender Systems

P11 Kowald, D., Schedl, M., Lex, E. (2020). The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In *Proceedings of the 42nd European Conference on Information Retrieval (ECIR'2020)*, pp. 35-42.
DOI: https://doi.org/10.1007/978-3-030-45442-5_5



The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study

Dominik Kowald^{1(✉)}, Markus Schedl², and Elisabeth Lex³

¹ Know-Center GmbH, Graz, Austria

dkowald@know-center.at

² Johannes Kepler University Linz, Linz, Austria

markus.schedl@jku.at

³ Graz University of Technology, Graz, Austria

elisabeth.lex@tugraz.at

Abstract. Research has shown that recommender systems are typically biased towards popular items, which leads to less popular items being underrepresented in recommendations. The recent work of Abdollahpouri et al. in the context of movie recommendations has shown that this popularity bias leads to unfair treatment of both long-tail items as well as users with little interest in popular items. In this paper, we reproduce the analyses of Abdollahpouri et al. in the context of music recommendation. Specifically, we investigate three user groups from the Last.fm music platform that are categorized based on how much their listening preferences deviate from the most popular music among all Last.fm users in the dataset: (i) low-mainstream users, (ii) medium-mainstream users, and (iii) high-mainstream users. In line with Abdollahpouri et al., we find that state-of-the-art recommendation algorithms favor popular items also in the music domain. However, their proposed Group Average Popularity metric yields different results for Last.fm than for the movie domain, presumably due to the larger number of available items (i.e., music artists) in the Last.fm dataset we use. Finally, we compare the accuracy results of the recommendation algorithms for the three user groups and find that the low-mainstreaminess group significantly receives the worst recommendations.

Keywords: Algorithmic fairness · Recommender systems · Popularity bias · Item popularity · Music recommendation · Reproducibility

1 Introduction

Recommender systems are quintessential tools to support users in finding relevant information in large information spaces [10]. However, one limitation of typical recommender systems is the so-called popularity bias, which leads to the underrepresentation of less popular (i.e., long-tail) items in the recommendation

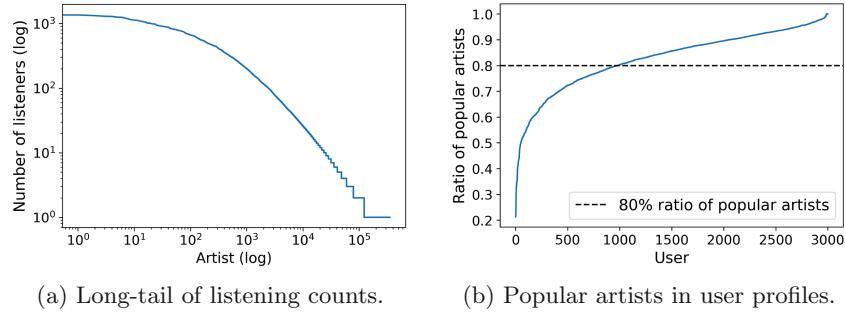


Fig. 1. Listening distribution of music artists. We find that around 1/3 (i.e., 1,000) of our users actually listen to at least 20% of unpopular artists.

lists [1,4,5]. The recent work of Abdollahpouri et al. [2] has investigated this popularity bias from the user perspective in the movie domain. The authors have shown that state-of-the-art recommendation algorithms tend to underserve users, who like unpopular items.

In this paper, we reproduce this study and conduct it in the music domain. As described in [16], there are several aspects of music recommendations that make them different to, e.g., movie recommendations such as the vast amount of available items. Therefore, we investigate music recommendations concerning popularity bias and, for reasons of comparability, raise the same two research questions as in [2]:

- *RQ1*: To what extent are users or groups of users interested in popular music artists?
- *RQ2*: To what extent does the popularity bias of recommendation algorithms affect users with different inclination to mainstream music?

For our experiments, we use a publicly available Last.fm dataset and address *RQ1* in Sect. 2 by analyzing the popularity of music artists in the user profiles. Next, we address *RQ2* in Sect. 3 by comparing six state-of-the-art music recommendation algorithms concerning their popularity bias propagation.

2 Popularity Bias in Music Data

For our reproducibility study, we use the freely available LFM-1b dataset [14]. Since this dataset contains 1.1 billion listening events of more than 120,000 Last.fm users and thus is much larger than the MovieLens dataset used in [2], we focus on a subset of it. Precisely, we extract 3,000 users that reflect the three user groups investigated in [2]. To this end, we use the mainstreamness score, which is available for the users in the LFM-1b dataset and which is defined as the overlap between a user’s listening history and the aggregated listening history of all Last.fm users in the dataset [3]. It thus represents a proxy for a user’s inclination to popular music.

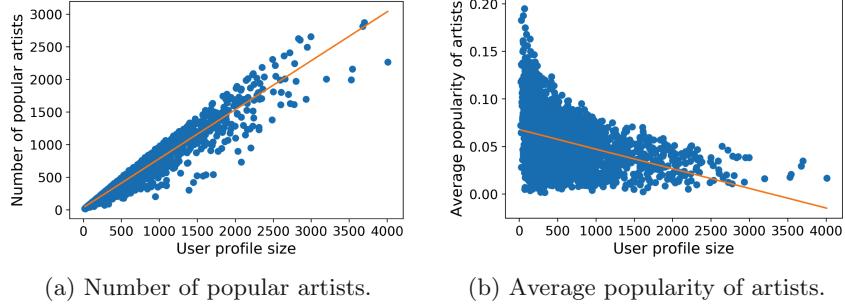


Fig. 2. Correlation of user profile size and the popularity of artists in the user profile. While there is a positive correlation between profile size and number of popular artists, there is a negative correlation between profile size and the average artist popularity.

Our subset consists of the 1,000 users with lowest mainstreamness scores (i.e., the LowMS group), the 1,000 users with a mainstreamness score around the median (i.e., the MedMS group), and the 1,000 users with the highest mainstreamness scores (i.e., the HighMS group). In total, we investigate 1,755,361 user-artist interactions between 3,000 users and 352,805 music artists. Compared to the MovieLens dataset with only 3,900 movies that Abdollahpouri et al. [2] have used in their study, our itemset is, consequently, much larger.

Listening Distribution of Music Artists. Fig. 1 depicts the listening distribution of music artists in our Last.fm dataset. As expected, in Fig. 1a, we observe a long-tail distribution of the listener counts of our items (i.e., artists). That is, only a few artists are listened to by many users, while most artists (i.e., the long-tail) are only listened to by a few users. Furthermore, in Fig. 1b, we plot the ratio of popular artists in the profiles of our 3,000 Last.fm users. As in [2], we define an artist as popular if the artist falls within the top 20% of artists with the highest number of listeners. We see that around 1,000 of our 3,000 users (i.e., around 1/3) have at least 20% of unpopular artists in their user profiles. This number also corresponds to the number of low-mainstream users we have in the LowMS user group.

User Profile Size and Popularity Bias in Music Data. Next, in Fig. 2, we investigate if there is a correlation between the user profile size (i.e., number of distinct items/artists) and the popularity of artists in the user profile. Therefore, in Fig. 2a, we plot the number of popular artists in the user profile over the profile size. As expected, we find a positive correlation ($R = .965$) since the likelihood of having popular artists in the profile increases with the number of items in the profile. However, when plotting the average popularity of artists in the user profile over the profile size in Fig. 2b, we find a negative correlation ($R = -.372$), which means that users with a smaller profile size tend to listen to more popular artists. As in [2], we define the popularity of an artist as the ratio of users who have listened to this artist.

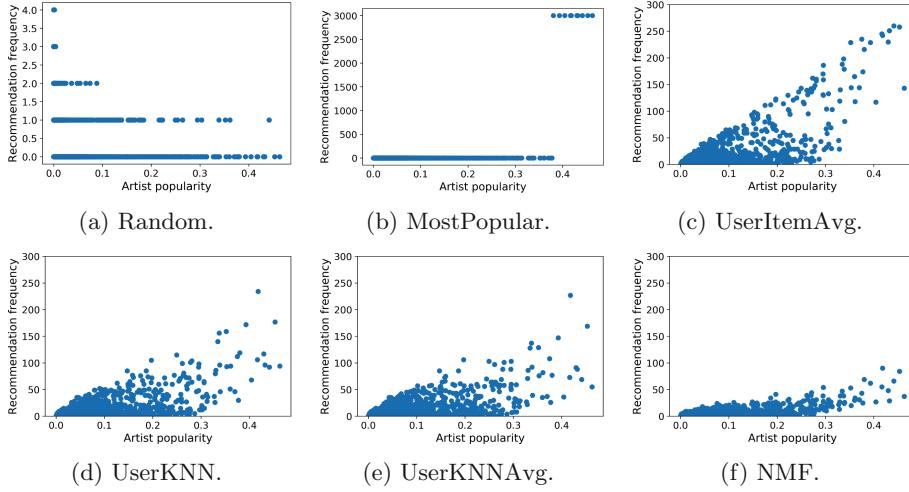


Fig. 3. Correlation of artist popularity and recommendation frequency. For all six algorithms, the recommendation frequency increases with the artist popularity.

Concerning *RQ1*, we find that one-third of our Last.fm users have at least 20% of unpopular artists in their profiles and thus, are also interested in low-mainstream music. Furthermore, we find that users with a small profile size tend to have more popular artists in their profiles than users with a more extensive profile size. These findings are in line with what Abdollahpouri et al. have found [2].

3 Popularity Bias in Music Recommendation

In this section, we study popularity bias in state-of-the-art music recommendation algorithms. To foster the reproducibility of our study, we calculate and evaluate all recommendations with the Python-based open-source recommendation toolkit Surprise¹. Using Surprise, we formulate our music recommendations as a rating prediction problem, where we predict the preference of a target user u for a target artist a . We define the preference of a for u by scaling the listening count of a by u to a range of [0, 1000] as also done in [15]. We then recommend the top-10 artists with the highest predicted preferences.

Recommendation of Popular Music Artists. We use the same evaluation protocol (i.e., 80/20 train/test split) and types of algorithms as in [2], which includes (i) baseline approaches, (ii) KNN-based approaches, and (iii) Matrix Factorization-based approaches. Specifically, we evaluate three baselines, i.e., Random, MostPopular, and UserItemAvg, which predicts the average listening count in the dataset by also accounting for deviations of u and a (e.g., if u tends

¹ <http://surpriselib.com/>.

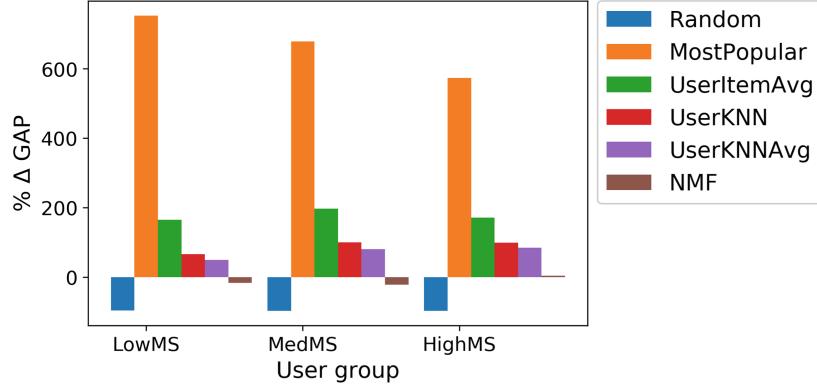


Fig. 4. Group Average Popularity (Δ GAP) of recommendation algorithms for LowMS, MedMS and HighMS. Except for the Random and NMF algorithms, all approaches provide too popular artist recommendations for all three user groups.

to have in general more listening events than the average Last.fm user) [6]. We also evaluate the two KNN-based approaches [13] UserKNN and UserKNNAvg, which is a hybrid combination of UserKNN and UserItemAvg. Finally, we include NMF (Non-Negative Matrix Factorization) into our study [9]. To reduce the computational effort of our study, in our evaluation, we exclude ItemKNN [12] as well as SVD++ [11] in contrast to [2]. In Fig. 3, we plot the correlation of artist popularity and how often the six algorithms recommend these artists. For all algorithms except for Random, we find a positive correlation, which means that popular items are recommended more often than unpopular items. As expected, this effect is most evident for the MostPopular algorithm and not present at all for the Random algorithm. It also seems that this popularity bias is not as strong in the case of NMF, which we will investigate further in the next section of this paper.

Popularity Bias for Different User Groups. To investigate the popularity bias of music recommendations for different user groups (i.e., LowMS, MedMS, and HighMS), we use the Group Average Popularity (GAP) metric proposed in [2]. Here, $GAP(g)_p$ measures the average popularity of the artists in the user profiles p of a specific user group g . We also define $GAP(g)_r$, which measures the average popularity of the artists recommended by a recommendation algorithm r to the users of group g . For each algorithm and user group, we are interested in the change in GAP (i.e., ΔGAP), which shows how the popularity of the recommended artists differs from the expected popularity of the artists in the user profiles. Hence, $\Delta GAP = 0$ would indicate fair recommendations in terms of item popularity, where fair means that the average artist popularity of the recommendations a user receives matches the average artist popularity in the user's profile. It is given by: $\Delta GAP = \frac{GAP(g)_r - GAP(g)_p}{GAP(g)_p}$.

Table 1. MAE results (the lower, the better) for four personalized recommendation algorithms and our three user groups. The worst (i.e., highest) results are always given for the LowMS user group (statistically significant according to a t-test with $p < .005$ as indicated by ***). Across the algorithms, the best (i.e., lowest) results are provided by NMF (indicated by bold numbers).

User group	UserItemAvg	UserKNN	UserKNNAvg	NMF
LowMS	42.991***	49.813***	46.631***	38.515***
MedMS	33.934	42.527	37.623	30.555
HighMS	40.727	46.036	43.284	37.305
All	38.599	45.678	41.927	34.895

In Fig. 4, we plot the ΔGAP for our six algorithms and three user groups. In contrast to the results presented in [2], where the LowMS group (i.e., the niche users) receives the highest values, we do not observe a clear difference between the groups except for MostPopular. We think that this is the case because of the large number of items we have in our Last.fm dataset (i.e., 352,805 artists compared to 3,900 movies in MovieLens). However, in line with Fig. 3, we again find that Random and NMF provide the fairest recommendations.

To further investigate *RQ2*, we analyze the Mean Average Error (MAE) [17] results of the four personalized algorithms for our user groups. As shown in Table 1, the LowMS group receives significantly worse (according to a t-test) recommendations than MedMS and HighMS for all algorithms. Interestingly, the MedMS group gets the best recommendations, probably since the users in this group have the largest profiles (i.e., on average 715 artists per user as compared to around 500 for the other two groups). Across the algorithms, NMF provides the best results. This is especially of interest since NMF also provided the fairest results in terms of artist popularity across the personalized algorithms.

4 Conclusion and Future Work

In this paper, we reproduced the study of [2] on the unfairness of popularity bias in movie recommender systems, which we adopted to the music domain. Similar to the original paper, we find (i) that users only have a limited interest in popular items (*RQ1*) and (ii) that users interested in unpopular items (i.e., LowMS) receive worse recommendations than users interested in popular items (i.e., HighMS). However, we also find that the proposed GAP metric does not provide the same results for Last.fm as it does for MovieLens, probably due to the high number of available items.

For future work, we plan to adapt this GAP metric in order to make it more robust for various domains. Furthermore, we want to study the characteristics of the LowMS users in order to better understand why they receive the worst recommendations and to potentially overcome this with novel algorithms (e.g., [7]).

Reproducibility and Acknowledgements. We provide our Last.fm dataset samples via Zenodo² [8] and our source code with all used parameter settings via Github³. This work was funded by the Know-Center GmbH (FFG COMET program) and the H2020 projects TRIPLE (GA: 863420) and AI4EU (GA: 825619).

References

1. Abdollahpouri, H., Burke, R., Mobasher, B.: Controlling popularity bias in learning-to-rank recommendation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 42–46. ACM (2017)
2. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation. In: Workshop on Recommendation in Multi-stakeholder Environments (RMSE 2019), in conjunction with the 13th ACM Conference on Recommender Systems, RecSys 2019 (2019)
3. Bauer, C., Schedl, M.: Global and country-specific mainstreaminess measures: definitions, analysis, and usage for improving personalized music recommendation systems. PloS one **14**(6), e0217389 (2019)
4. Brynjolfsson, E., Hu, Y.J., Smith, M.D.: From niches to riches: anatomy of the long tail. Sloan Manag. Rev. **47**(4), 67–71 (2006)
5. Jannach, D., Lerche, L., Kamehkhosh, I., Jugovac, M.: What recommenders recommend: an analysis of recommendation biases and possible countermeasures. User Model. User-Adap. Interact. **25**(5), 427–491 (2015). <https://doi.org/10.1007/s11257-015-9165-3>
6. Koren, Y.: Factor in the neighbors: scalable and accurate collaborative filtering. ACM Trans. Knowl. Discov. Data (TKDD) **4**(1), 1 (2010)
7. Kowald, D., Lex, E., Schedl, M.: Modeling artist preferences for personalized music recommendations. In: Proceedings of the Late-Breaking-Results Track of the 20th Annual Conference of the International Society for Music Information Retrieval (ISMIR 2019) (2019)
8. Kowald, D., Schedl, M., Lex, E.: LFM user groups (2019). <https://doi.org/10.5281/zenodo.3475975>
9. Luo, X., Zhou, M., Xia, Y., Zhu, Q.: An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. IEEE Trans. Ind. Inform. **10**(2), 1273–1284 (2014)
10. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer, Boston (2011). https://doi.org/10.1007/978-0-387-85820-3_1
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Incremental singular value decomposition algorithms for highly scalable recommender systems. In: Proceedings of the Fifth International Conference on Computer and Information Science, vol. 27, p. 28 (2002)
12. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J., et al.: Item-based collaborative filtering recommendation algorithms. In: WWW 1, 285–295 (2001)

² <https://doi.org/10.5281/zenodo.3475975>.

³ <https://github.com/domkowald/LFM1b-analyses>.

13. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 291–324. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_9
14. Schedl, M.: The LFM-1B dataset for music retrieval and recommendation. In: Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval (ICMR 2016), pp. 103–110. ACM, New York (2016)
15. Schedl, M., Bauer, C.: Distance-and rank-based music mainstreaminess measurement. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, pp. 364–367. ACM (2017)
16. Schedl, M., Zamani, H., Chen, C.-W., Deldjoo, Y., Elahi, M.: Current challenges and visions in music recommender systems research. *Int. J. Multimedia Inf. Retr.* **7**(2), 95–116 (2018). <https://doi.org/10.1007/s13735-018-0154-2>
17. Willmott, C.J., Matsuura, K.: Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.* **30**(1), 79–82 (2005)

P12 Support the Underground: Characteristics of Beyond-Mainstream Music Listeners (2021)

Fairness and Popularity Bias in Recommender Systems

P12 Kowald, D., Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E. (2021). Support the Underground: Characteristics of Beyond-Mainstream Music Listeners. *EPJ Data Science*, 10:14.

DOI: <https://doi.org/10.1140/epjds/s13688-021-00268-9>



REGULAR ARTICLE

Open Access



Support the underground: characteristics of beyond-mainstream music listeners

Dominik Kowald^{1*} , Peter Muellner¹ , Eva Zangerle² , Christine Bauer³ , Markus Schedl^{1,4,5} and Elisabeth Lex^{6*}

*Correspondence:
dkowald@know-center.at;
elisabeth.lex@tugraz.at
¹Know-Center GmbH, Graz, Austria
⁶Graz University of Technology,
Graz, Austria
Full list of author information is
available at the end of the article

Abstract

Music recommender systems have become an integral part of music streaming services such as Spotify and Last.fm to assist users navigating the extensive music collections offered by them. However, while music listeners interested in mainstream music are traditionally served well by music recommender systems, users interested in music beyond the mainstream (i.e., non-popular music) rarely receive relevant recommendations. In this paper, we study the characteristics of beyond-mainstream music and music listeners and analyze to what extent these characteristics impact the quality of music recommendations provided. Therefore, we create a novel dataset consisting of Last.fm listening histories of several thousand beyond-mainstream music listeners, which we enrich with additional metadata describing music tracks and music listeners. Our analysis of this dataset shows four subgroups within the group of beyond-mainstream music listeners that differ not only with respect to their preferred music but also with their demographic characteristics. Furthermore, we evaluate the quality of music recommendations that these subgroups are provided with four different recommendation algorithms where we find significant differences between the groups. Specifically, our results show a positive correlation between a subgroup's openness towards music listened to by members of other subgroups and recommendation accuracy. We believe that our findings provide valuable insights for developing improved user models and recommendation approaches to better serve beyond-mainstream music listeners.

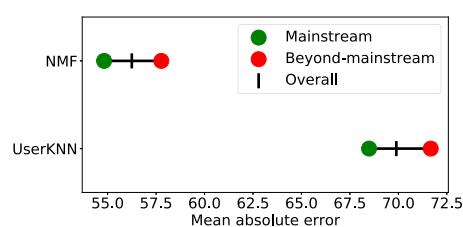
Keywords: Music recommender systems; Acoustic features; Last.fm; Clustering; User modeling; Fairness; Popularity bias; Beyond-mainstream users

1 Introduction

In the digital era, users have access to continually increasing amounts of music via music streaming services such as Spotify and Last.fm. Music recommender systems have become an essential means to help users deal with content and choice overload as they assist users in searching, sorting, and filtering these extensive music collections. Simultaneously, both music listeners and artists benefit from the employed segmentation and personalization approaches that are typically leveraged in music recommendation approaches [1]. As a result, users with different preferences and needs can be targeted in various ways with the

© The Author(s) 2021. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Figure 1 Recommendation accuracy measured by the mean absolute error (MAE) of a non-negative matrix factorization-based approach (i.e., NMF [10]) and a neighborhood-based approach (i.e., UserKNN [11]) for mainstream and beyond-mainstream user groups in Last.fm. We see that beyond-mainstream users receive a substantially lower recommendation quality (i.e., higher MAE) compared to mainstream music listeners. Thus, for recommender systems, it is harder to provide high-quality recommendations to beyond-mainstream than to mainstream listeners



goal that all users are presented the information and content that they need or prefer. This also means that current recommendation techniques should serve all users equally well, independent of their inclination to popular content.

Present work In the paper at hand, we focus on music consumers who listen to music beyond the mainstream (i.e., users who listen to non-popular music) in the music streaming platform Last.fm.¹ As highlighted in Fig. 1, current recommender systems do not work well for consumers of beyond-mainstream music (see Sect. 3.5 for details on this analysis). In contrast, music consumers who listen to popular music seem to get better recommendations. This finding is not essentially new. In fact, it is a widely-known problem that recommender systems (and those based on collaborative filtering, in particular) are prone to popularity bias, which leads to the behavior that long-tail items (i.e., items with few user interactions) have little chance being recommended. This phenomenon is also present across different application domains such as movies [2] or music [3].

Our previous work [4] has shown that users interested in beyond-mainstream music tend to have larger user profile sizes (i.e., individual users show a high(er) number of distinct artists they have listened to) compared to users interested in mainstream music. The observation that beyond-mainstream music listeners produce a substantial amount of digital footprints motivates the need to improve the recommendation quality for this group. However, although related research has already studied the long-tail recommendation problem (e.g., [5–8]; see Sect. 2 for a more detailed discussion of related work), it is still a fundamental challenge to understand and identify the characteristics of beyond-mainstream music and beyond-mainstream music listeners. Additionally, related work [9] has shown that the group-specific concepts of openness and diversity influence recommendation quality, where openness is defined as across-group diversity (i.e., do users of one group listen to the music of other groups?) and diversity is defined as within-group variability (i.e., how dissimilar is the music listened to by users within groups?). Thus, we are also interested in the correlation between the characteristics of beyond-mainstream music and music listeners with openness and diversity patterns as well as with recommendation quality. Concretely, our work is guided by the following research question:

RQ: What are the characteristics of beyond-mainstream music tracks and music listeners, and how do these characteristics correlate with openness and diversity patterns as well as with recommendation quality?

¹<https://www.last.fm/>

To address this research question, we create, provide, and analyze a novel dataset called *LFM-BeyMS*, which contains complete listening histories of more than 2000 beyond-mainstream music listeners mined from the Last.fm music streaming platform. Besides, our dataset is enriched with acoustic features and genres of music tracks. Using this enriched dataset, we identify different types of beyond-mainstream music via unsupervised clustering applied to the acoustic features of music tracks. We then characterize the resulting music clusters using music genres. Then, we assign beyond-mainstream users to the clusters to further divide the beyond-mainstream users into subgroups. We study how the characteristics of these beyond-mainstream subgroups correlate with openness and diversity patterns as well as with recommendation quality measured through prediction accuracy.

Findings and contributions We identify four clusters of beyond-mainstream music in our dataset: (i) C_{folk} , music with high acousticness such as “folk”, (ii) C_{hard} , high energy music such as “hardrock”, (iii) C_{ambi} , music with high acousticness and high instrumentalness such as “ambient”, and (iv) C_{elec} , music with high energy and high instrumentalness such as “electronica”. By assigning users to these clusters, we get four distinct subgroups of beyond-mainstream music listeners: (i) U_{folk} , (ii) U_{hard} , (iii) U_{ambi} , and (iv) U_{elec} . We also find that these groups differ considerably with respect to the accuracy of recommendations they receive, where group U_{ambi} gets significantly better recommendations than U_{hard} . When relating our results to openness and diversity patterns of the subgroups, we find that U_{ambi} is the most open but least diverse group, while U_{hard} is the least open but most diverse group. This is in line with related research [9], which has shown that openness is stronger correlated with accurate recommendations than diversity. This means that users are more likely to accept recommendations from different groups (i.e., openness) rather than varied within a group (i.e., diversity).

Summed up, our contributions are:

- We identify more than 2000 beyond-mainstream music listeners on the Last.fm platform and enrich their listening profiles with acoustic features and genres of music tracks listened to (Sects. 3.1–3.4).
- We validate related research by showing that beyond-mainstream music listeners receive a significantly lower recommendation accuracy than mainstream music listeners (Sect. 3.5).
- We identify four clusters of beyond-mainstream music using unsupervised clustering and characterize them with respect to acoustic features and music genres (Sect. 4.1).
- We define four subgroups of beyond-mainstream music listeners by assigning users to the music clusters and discuss the relationship between openness, diversity, and recommendation quality of these groups (Sect. 4.2).
- To foster reproducibility of our research, we make available our novel *LFM-BeyMS* dataset via Zenodo² and the entire Python-based implementation of our analyses via Github.³

We believe that our findings provide useful insights for creating user models and recommendation algorithms that better serve beyond-mainstream music listeners.

²<https://doi.org/10.5281/zenodo.3784764>

³<https://github.com/pmuellner/supporttheunderground>

2 Related work

We identify three strands of research that are relevant to our work: (i) modeling of music preferences, (ii) long-tail recommendations, and (iii) popularity bias in music recommender systems.

Modeling of music preferences A multitude of factors [12] influences musical tastes and musical preferences of users. Characteristics of music listeners and music preferences have been studied in various research domains [13], ranging from music sociology [14] and psychology [15] to music information retrieval and music recommender systems [1]. Studies on music listening behavior showed that personal traits and long-term music preferences are correlated as people tend to prefer music styles that align with their personalities [16, 17]. Furthermore, related work found a relationship between music and motivation [18], music and emotion [19–22] or both personality and emotion [23]. Openness, a personality trait from the Five Factor Model [24], has also been shown to positively influence a user's preference for music recommendations [9]. Specifically, the authors of [9] found that people tend to prefer recommendations from different kinds of music (i.e., openness) rather than varied within a specific kind of music (i.e., diversity). Others showed that familiarity has a positive influence on music preferences [25, 26] and that music preferences may change over time [27]. Another strand of research on modeling users' music preferences leverages content features, e.g., acoustic features. It has been shown that the distribution of acoustic features of a user's preferred genre substantially influences the user's choice of music within other genres [28]. Also, acoustic features have been utilized to model users' preferences under different contextual conditions, in order to refine recommendation quality [29]. Based on tracks' acoustic features, the authors of [30] identified several types of music, and subsequently modeled each user by linearly combining the acoustic features of the music types. In contrast to these works, we focus on using acoustic features of music tracks for modeling and clustering beyond-mainstream music. Additionally, we link these beyond-mainstream music clusters to music genres and users in our Last.fm data sample.

Long-tail recommendations Related research [6, 7] has found that individual music consumption is biased towards popular music and that usage data for less popular music is scarce. Due to the scarcity problem, items with no or few ratings (i.e., long-tail items) have little chance of being recommended [5]. As a consequence, users that particularly favor items with few ratings or interactions are less likely to be recommended those items that they like [3]. That is problematic because many users, from time to time, prefer niche music [8]. Therefore, such users are not well served as a result of their preference for less popular items. That has been attributed to *popularity bias*, which corresponds to over-representation of popular items in the recommendation lists [31–33]. Abdollahpouri et al. [2] studied popularity bias in a dataset of movies (i.e., the MovieLens 1M dataset [34]) from the user perspective. Their study showed that commonly used recommendation techniques tend to deliver worse recommendations to users who prefer less popular movies. In our work [4], we found evidence for popularity bias in a Last.fm dataset and showed that traditional personalized recommendation algorithms such as collaborative filtering deliver worse recommendations for consumers of niche music. In the present work, we aim to gain a deeper understanding of the behavior and preferences of this

beyond-mainstreaminess user group. Thus, in contrast to existing works in long-tail recommendations, we focus on the user rather than the item perspective.

Popularity bias in music recommender systems Music recommender systems [1] are crucial tools in online streaming services such as Last.fm, Pandora, or Spotify. They help users find music that is tailored to their preferences. The basis of music recommender systems are user models derived from users' listening behavior, user properties such as personality (e.g., [35]), content features of music, or hybrid combinations of both, e.g., [36–39]. As discussed earlier, due to insufficient amounts of usage data for less popular items, many music recommendation algorithms do not provide useful recommendations for consumers of less popular and niche items. As a remedy, in [40], an approach is suggested that divides music consumers into experts and novices according to their long tail distribution in their playlists. These experts are then converted to nodes with bidirectional links connecting all the experts. These links are created to perform link analysis on the graph and to assign fine-grained weights to songs. The presented approach helps add music from the long-tail into the recommendation list. In our previous research [41, 42], we have used a framework [43] that employs insights from human memory theory to design a music recommendation algorithm that provides more accurate recommendations than collaborative filtering-based approaches for three groups of users, i.e., low-mainstream, medium-mainstream and high-mainstream users. While the awareness of popularity bias in music recommender systems increases (e.g., [44]), the characteristics of music consumers whose preferences lie beyond popular, mainstream music are still not well understood. In the present work, we shed light on the characteristics of such beyond-mainstream music consumers and relate them to openness and diversity patterns as well as recommendation quality. With this, we aim to provide useful insights for creating novel music recommendation models that mitigate popularity bias.

3 Preliminaries

We investigate the characteristics of beyond-mainstream music listeners in a dataset mined from Last.fm, a popular music streaming platform. We characterize the tracks in our dataset with acoustic features. Besides, we compare the recommendation accuracy of beyond-mainstream music listeners with the one of mainstream music listeners to motivate our subsequent analysis of the characteristics of beyond-mainstream music listeners.

3.1 Acoustic music features

For our analyses, we characterize music tracks using acoustic features that describe the content of a given track. Following the lines of, e.g., [30, 45–47], we rely on acoustic features provided by the Spotify API as a compact characterization of tracks.⁴ The following eight features are extracted from the audio signal of a track:

Danceability captures how suitable a track is for dancing and is computed based “on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity”.

Energy describes the perceived intensity and activity of a track and is based on the dynamic range, perceived loudness, timbre, onset rate, and general entropy of a track.

⁴<https://developer.spotify.com/web-api/get-several-audio-features/>

Speechiness captures the presence of spoken words in a track. High speechiness values indicate a high degree of spoken words (e.g., an audiobook), whereas medium values indicate tracks with both music and speech (e.g., rap music). Low values represent typical music tracks.

Acousticness measures the probability that the given track only contains acoustic instruments.

Instrumentalness quantifies the probability that a track contains no vocals, i.e., the track is instrumental.

Tempo measures the rate of the track's beat in beats per minute.

Valence describes the “emotional positiveness” conveyed by a track (i.e., cheerful and euphoric tracks reach high valence values).

Liveness measures the probability that a track was performed live, i.e., whether an audience is present in the recording.

3.2 Enriched dataset of music listening events

To study characteristics of beyond-mainstream users and their listening preferences, we create a novel dataset called *LFM-BeyMS* that contains the required information for such analyses. We base our work on a dataset gathered from the Last.fm music platform, which we considerably enrich with the music tracks’ acoustic features (see Sect. 3.1) [48]. Additionally, we combine this data with mainstreaminess information of Last.fm users (see Sect. 3.3) as well as music genre information to identify beyond-mainstream listeners and music (see Sect. 3.4).

An overview of our new *LFM-BeyMS* dataset and its data sources is depicted in Fig. 2.⁵ As shown, the starting point for our new dataset is the publicly available *LFM-1b* dataset⁵ of music listening information shared by users of the online music platform Last.fm [49]. *LFM-1b* contains listening histories of 120,322 users; their listening records (or “listening events”) have been created between January 2005 and August 2014. They sum up to over 1.1 billion listening events (LEs), where each LE is described by an (anonymous) user identifier, the artist name, the album name, the track name, and the timestamp of the listening event. Also, the *LFM-1b* dataset includes demographics of some users (i.e., country, age, and gender).

To enrich the *LFM-1b* dataset to suit our task, we utilize our previously created *CultMRS* music recommendation dataset [50]. This dataset contains 55,191 users, who have listened to a total of 26,022,625 distinct tracks, captured by a total of 807,890,921 LEs [48].

To further enrich the dataset with music acoustic features, we gather the acoustic features described in Sect. 3.1 for the tracks remaining in the dataset after the filtering described above. To this end, we rely on the Spotify API to gather content-based acoustic features for each track. Particularly, we search tracks using the \langle track, artist, album \rangle triples extracted from the *LFM-1b* dataset using the Spotify search API⁶ to gather the Spotify track URI of each track by using all three parts of the triple in a conjunctive query. In total, this allowed gathering 4,326,809 Spotify URIs. For the remainder of the tracks, we were not able to retrieve a URI. We attribute this to two factors: either the searched track is not provided by Spotify or the track, artist, and album information cannot be matched to

⁵<http://www.cp.jku.at/datasets/LFM-1b/>

⁶<https://developer.spotify.com/web-api/search-item/>

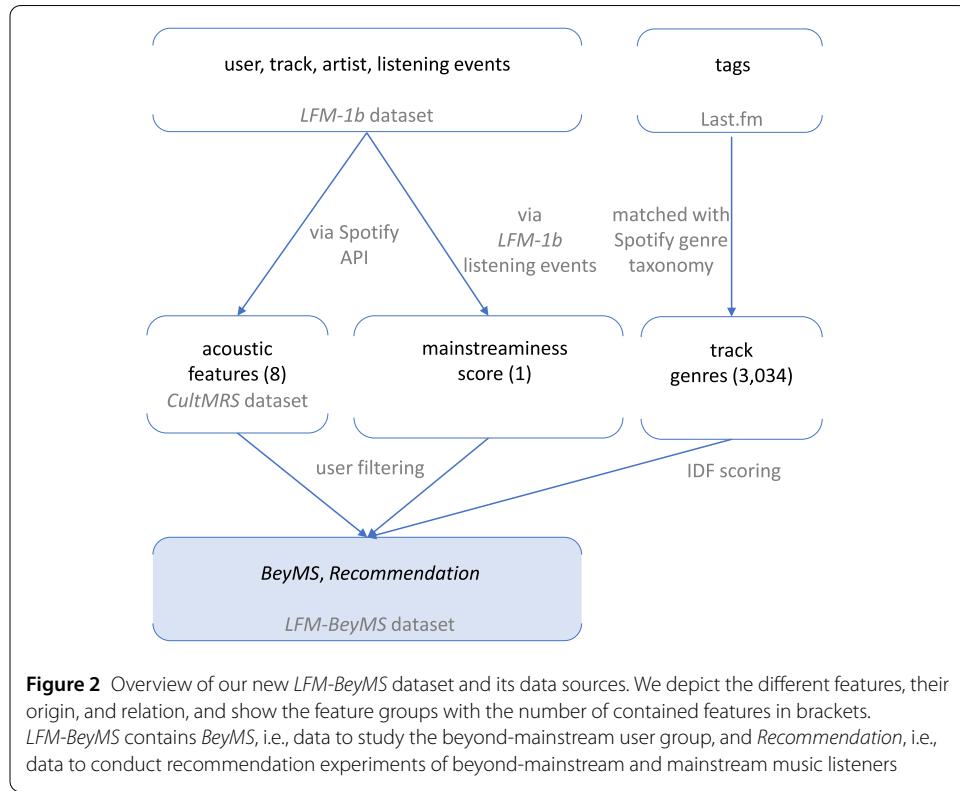


Figure 2 Overview of our new *LFM-BeyMS* dataset and its data sources. We depict the different features, their origin, and relation, and show the feature groups with the number of contained features in brackets. *LFM-BeyMS* contains *BeyMS*, i.e., data to study the beyond-mainstream user group, and *Recommendation*, i.e., data to conduct recommendation experiments of beyond-mainstream and mainstream music listeners

a Spotify track unambiguously. Subsequently, we use the obtained track URI to query the acoustic features API,⁷ which returns the acoustic features of a given track (cf. Sect. 3.1). In a subsequent cleaning step, we remove all tracks for which the Spotify API did not provide the full set of acoustic features.

That procedure provides us with a set of 3,478,399 unique tracks and their acoustic features. Within the LFM-1b dataset, this amounts to 13.36% of the distinct tracks. Overall, these account for as much as 48.89% of all listening events (i.e., the tracks listened to by users) of the LFM-1b dataset. The resulting dataset, now enriched by acoustic music descriptors, comprises a total of approximately 394 million listening events of 55,149 users. In Table 1 (column “*CultMRS*”), we provide further descriptive statistics of the *CultMRS* dataset. We refine this dataset to create our new *LFM-BeyMS* dataset (column “*BeyMS*” in Table 1), which consists of *BeyMS*, i.e., data to study the characteristics of beyond-mainstream music listeners, and *Recommendation*, i.e., data to conduct recommendation experiments of beyond-mainstream and mainstream music listeners.

3.3 Identifying beyond-mainstream music listeners

To identify beyond-mainstream music listeners, for each user, we compute a mainstreaminess score, which is generally defined as the overlap between a user’s individual listening history and the aggregated listening history of all Last.fm users in the dataset. In this vein, the mainstreaminess score reflects a user’s inclination to music listened to by the Last.fm mainstream listeners (i.e., the “average” Last.fm listener in the dataset). In [51], several

⁷<https://developer.spotify.com/web-api/get-several-audio-features/>

Table 1 Descriptive statistics of the *CultMRS* dataset and our novel *LFM-BeyMS* dataset. *CultMRS* comprises acoustic features of tracks. *LFM-BeyMS* is based on *CultMRS* and consists of *BeyMS* and *Recommendation*. Our analyses of beyond-mainstream music listeners utilize *BeyMS* and our recommendation experiments utilize *Recommendation*, which includes listening events of both users with beyond-mainstream and mainstream music taste

Item	<i>CultMRS</i> [50]	<i>LFM-BeyMS</i> (our novel dataset)	
		<i>BeyMS</i>	<i>Recommendation</i>
Users	55,149	2074	4148
Tracks	3,478,399	157,444	1,084,922
Artists	337,840	14,922	110,898
Listening Events (LEs)	394,944,868	4,916,174	16,687,363
Min. LEs per user	1	3	9
Q_1 LEs per user	1442	1254	2604
Median LEs per user	5667	2048	3766
Q_3 LEs per user	9738	3239	5252
Max. LEs per user	399,210	10,536	11,177
Avg. LEs per user	7161.41 (\pm 10,326.91)	2371.526 (\pm 1520.629)	4,022.990 (\pm 1898.060)

measures of user mainstreaminess are defined. Out of these, we choose the *M-global-R-APC* definition since it yielded good results in context-based music recommendation experiments for the *LFM-1b* dataset, as evidenced in [51]. The *M-global-R-APC* measure approximates a user's mainstreaminess score by computing Kendall's τ [52] rank correlation between the user's vector of artist play counts and the global vector of artist play counts (aggregated over all users in the dataset). This definition also explains the name of the measure, where "M" refers to mainstreaminess, "global" indicates the global perspective, "R" stands for rank correlation, and "APC" refers to artist play counts.

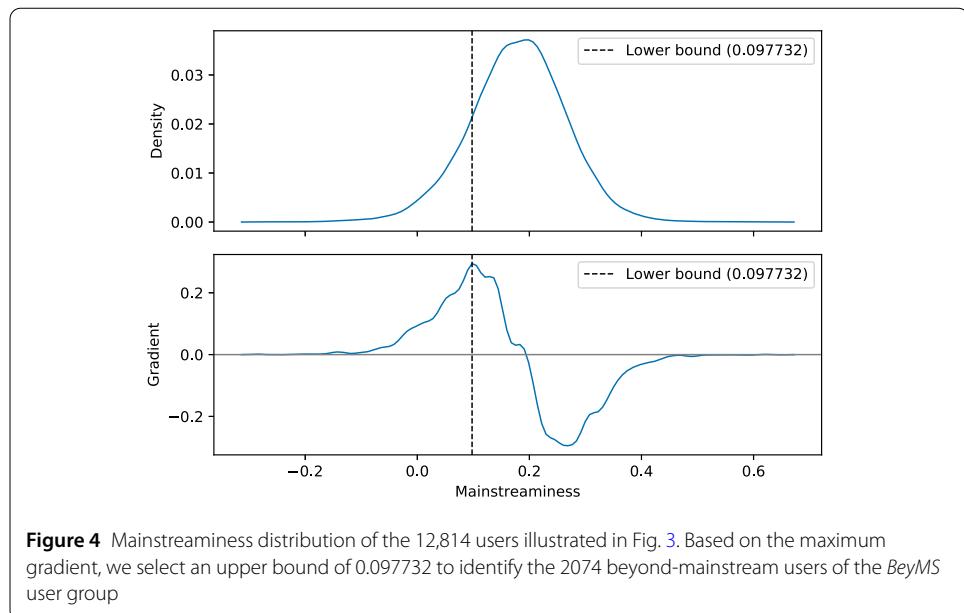
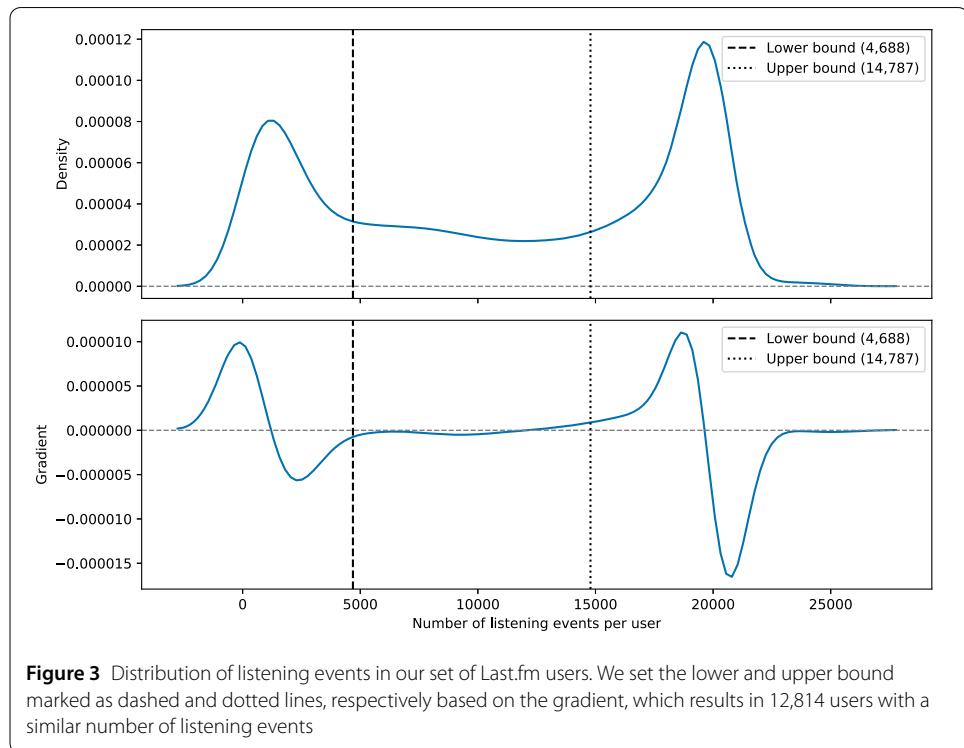
Next, we describe how we identify our beyond-mainstream users via filtering the users by the number of listening events (see Fig. 3 and Sect. 3.3.1) and by mainstreaminess scores (see Fig. 4 and Sect. 3.3.2).

3.3.1 Filtering users by the number of listening events

For our study, we select the users so that listeners of different levels of "listening activity" are equally represented. We conduct a Gaussian kernel density estimation (KDE) [53] on the distribution of listening events over users to estimate the continuous probability density function (PDF) [54]. However, KDE estimates the PDF via discrete bins and hence, it is necessary to approximate the gradient via the principle of finite differences. The gradient of the PDF helps us identifying regions of increasing or decreasing probability.

Figure 3 shows that two large subsets of users exist that exhibit either very few or an abundance of listening events. For our analysis, we consider only users who are not in one of the subsets as mentioned earlier. On the one hand, we exclude users with too little data available for studying their listening behavior; and on the other hand, we exclude so-called power listeners who might bias our analyses. Furthermore, such users with a very high number of listening events are often radio stations, which do not contribute reliable data to our investigations.

Hence, we define lower and upper bounds regarding the number of users' listening events to include in our study, such that the rate of change in terms of the number of listening events is minimal and stable within these boundaries. That requires the gradient of the region within the lower and upper bound to be near zero (i.e., $\pm 10^{-6}$). By computing the second-order accurate central differences [55], we obtain an approximation of the



gradient and find the longest cohesive region fulfilling the requirements between a lower bound of 4688 and an upper bound of 14,787 listening events per user, which leads to 12,814 users.

3.3.2 Filtering users by mainstreaminess scores

Figure 4 illustrates the mainstreaminess distribution of the 12,814 users that we have extracted based on the number of listening events. Here, mainstreaminess is defined accord-

ing to the *M-global-R-APC* definition taken from [51] (explained in Sect. 3.3). By setting an appropriate upper bound, we aim to exclude mainstream music listeners. In other words, we aim to set the upper bound to the beginning of the distribution's bulk, which is motivated as follows: Firstly, the first inflection point (i.e., maximal gradient) of a Gaussian distribution is found at $\mathbb{E}[X] - \text{std}(X)$, where $\mathbb{E}[X]$ is the expectation, and $\text{std}(X)$ is the standard deviation of the Gaussian random variable X . Secondly, the first inflection point of a Gaussian distribution is equivalent to the 15.9-percentile. By setting the mainstreamness threshold to this point, we intend to omit the majority of users and hence, only consider the 15.9% of users with the lowest mainstreamness scores. Utilizing this upper bound on the mainstreamness score, we obtain a set of 2074 beyond-mainstream users. Furthermore, the Gaussian assumption can be strengthened by the observation that the 2074 beyond-mainstream users represent 16.19% of users. In the remainder of this paper, we refer to this set of beyond-mainstream music listeners as *BeyMS*.

3.4 Identifying beyond-mainstream music

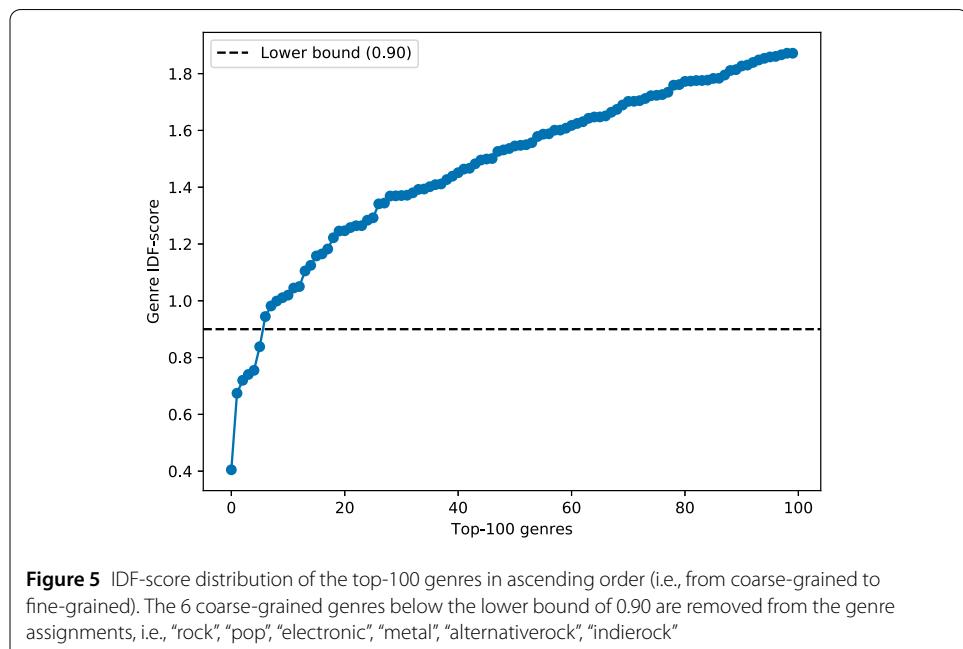
We aim to study beyond-mainstream listeners in terms of their music taste. We characterize music via its acoustic features, as described in Sect. 3.1, and also investigate genres as an alternative way to describe a music track via conventional categories. As the *LFM-1b* dataset does not contain genre annotations of tracks and the Spotify API only provides genres on artist level,⁸ we leverage the tags assigned to each track by Last.fm users to identify genre annotations. To obtain these tags, we use the respective Last.fm API endpoint.⁹ After having fetched the tags for each track, we de-capitalize them and remove all non-alpha-numeric characters. Since not all tags used by Last.fm users correspond to actual music genres (e.g., the “seenlive” tag is used to indicate that a user has seen an artist performing this track live), we use a fine-grained music genre taxonomy consisting of 3034 genres that are also utilized by Spotify, which we gather from the EveryNoise service (2019-07-24).¹⁰ Specifically, for each track listened to by any of our *BeyMS* users, we remove all tags that are not part of the EveryNoise genre taxonomy, using a case-insensitive matching approach.

We note that Last.fm users tend to assign very general genre tags to a large number of tracks, such as “pop” or “rock”. To remove these coarse-grained genres and to identify fine-grained beyond-mainstream music genres, we calculate the inverse document frequency (IDF) [56] metric of our genre-track distribution by treating genres as terms and tracks as documents, i.e., $\text{IDF}(g) = \log_{10} \frac{|T|}{|\{t \in T \text{ with } g \in G_t\}|}$. More precisely, the IDF-score of genre g is determined by relating the number of all tracks $|T|$ to the number of tracks annotated with genre g where $|G_t|$ is the set of genres assigned to track t . This way, a coarse-grained genre receives a small IDF-score, while a fine-grained genre receives a high IDF-score. Figure 5 shows the IDF-score distribution of the top-100 genres in ascending order (i.e., from coarse-grained to fine-grained). Here, we identify two groups of genres, where the first group consists of 6 genres with small IDF-scores, and the second group consists of 94 genres with high IDF-scores. The visual inspection of Fig. 5 indicates that the lower bound of 0.90 serves as a discriminant between these two groups of coarse-grained and

⁸<https://developer.spotify.com/documentation/web-api/reference-beta/#endpoint-get-an-artist>

⁹<https://www.last.fm/api/show/track.getTopTags>

¹⁰<http://everynoise.com/>



fine-grained genres. Consequently, we remove the 6 coarse-grained genres (i.e., “rock”, “pop”, “electronic”, “metal”, “alternativerock”, “indierock”) from the genre assignments of our tracks, which leads to 157,444 out of 799,659 tracks listened to by *BeyMS* users with at least one remaining genre. In total, these tracks are annotated with 1418 unique genre identifiers.

We are aware of the fact that our track filtering procedure leads to incomplete listening profiles of users. Since we rely on genres to describe beyond-mainstream music, these filtering steps are necessary for our study. To ensure that the *BeyMS* users’ reduced listening profiles are still representative of their music preferences, we further investigate the consequences of the filtering procedure. Here, we find that a user’s listening history (i.e., the entirety of a user’s listening events) is reduced to 61% on average. However, we also find that there are only 62 of the 2074 *BeyMS* users, for whom the listening history is reduced to less than 20%. For these users most affected by the filtering, we compare the acoustic feature distributions of their listened tracks before and after the filtering steps, and find that filtering only marginally affects the acoustic feature distributions (i.e., average change in mean = 0.0098 ± 0.0148). This means that the acoustic feature distribution contained in the user’s profile is highly robust against the filtering. The statistics of *BeyMS* are summarized in column “*BeyMS*” in Table 1.

3.5 Recommendations for beyond-mainstream music listeners

In order to compare the recommendation accuracy of recommendations received by the users of our *BeyMS* group and by mainstream users, we construct a dataset consisting of *BeyMS*’s listening events and the listening events of an equally-sized group of mainstream users. Therefore, we define the *MS* user group as 2074 (i.e., the size of our *BeyMS* group) randomly-chosen users with a mainstreaminess score that is higher than the upper bound for low mainstreaminess, identified in Fig. 4. Furthermore, the *MS* users are also in between the lower and upper bounds for listening events identified in Fig. 3. As shown in

Table 1 (column “*Recommendation*”), the dataset used for the evaluation of recommendations contains data of 4148 distinct *BeyMS* and *MS* users, 1,084,922 distinct tracks, and 16,687,363 listening events.

We use the Python-based open-source recommendation library Surprise¹¹ to compute and evaluate recommendations. One advantage of using Surprise is that it provides built-in recommendation algorithms as well as a standardized evaluation pipeline, which enhances the reproducibility of our research. Since Surprise is focused on rating prediction, we formulate our music recommendation scenario also as a rating prediction problem, in which we predict the preference of a target user u for a target track t . As done in [57], we model the preference of t for u by scaling the play count (i.e., number of listening events) of t by u to a range of [1; 1000] using min-max normalization. We perform this normalization on the individual user level to ensure that all users share the same preference value ranges. Thus, with this method, we ensure that each user’s most listened track has a preference value of 1000, while their least listened track has a preference value of 1. To ensure that this min-max normalization procedure does not disrupt the play count distribution of our users, we compare the original play count distribution with the normalized distribution and find that both distributions are strongly right-skewed. Specifically, we find very similar distributions for large amounts of our play count data.

We utilize a selection of Surprise’s built-in recommendation methods consisting of one baseline approach (i.e., UserItemAvg), two neighborhood-based approaches (i.e., UserKNN and UserKNNAvg), and one matrix factorization-based approach (i.e., NMF). Specifically, UserItemAvg predicts the average play count in the dataset by also accounting for deviations of u and t , for example, if a user u tends to have more listening events than the average Last.fm user [58]. UserKNN [11] is a user-based collaborative filtering approach and is calculated using $k = 40$ nearest neighbors and the cosine similarity metric, which are the default settings of Surprise. UserKNNAvg is an extension of UserKNN [11] that also takes the average rating of target user u into account. Finally, NMF, i.e., non-negative matrix factorization [10], is calculated using 15 latent factors, which is the default parameter in the Surprise library. As shown in our previous work [4], NMF is also capable of recommending non-popular items from the long tail and should therefore especially be of interest for our beyond-mainstream recommendation setting.

We use Surprise’s default parameters and refrain from performing any hyperparameter tuning since we are only interested in assessing (relative) performance differences between the two user groups *BeyMS* and *MS*, and not in outperforming any state-of-the-art algorithm. This is also the reason why we focus on traditional algorithms instead of investigating the most recent deep learning architectures, which would also require a much higher computational effort.

The resulting mean absolute error (MAE) results can be observed in Table 2 (and correspond to the ones already shown in Fig. 1). We favor MAE over the commonly used root mean squared error (RMSE) due to several pitfalls, especially regarding the comparison of groups with different numbers of observations [59]. Here, we perform 5-fold cross-validation leading to 5 different 80/20 train-test splits and average the MAE over the 5 folds. NMF clearly outperforms UserItemAvg as well as the two neighborhood-based methods (i.e., UserKNN and UserKNNAvg) both for the two user groups (see

¹¹<http://surpriselib.com/>

Table 2 Mean absolute error (MAE) results for the two user groups *MS* and *BeyMS* of different mainstreaminess and a selection of standard recommendation algorithms. A one-tailed Mann–Whitney-U test ($\alpha = 0.0001$) provides significant evidence, indicated by ***, that all algorithms perform worse on *BeyMS* than on *MS* in terms of MAE. Furthermore, NMF (as shown in bold) outperforms the other three approaches UserItemAvg, UserKNN and UserKNNAvg

User group	UserItemAvg	UserKNN	UserKNNAvg	NMF
<i>BeyMS</i>	63.4608***	71.6694***	67.5770***	57.7703***
<i>MS</i>	61.2562	68.4894	63.3985	54.8182
Overall	62.2315	69.8962	65.2469	56.2492

rows “*BeyMS*” and “*MS*”) separately and overall without distinguishing between the user groups (see row “Overall”). Additionally, we conduct a one-tailed Mann–Whitney-U test ($\alpha = 0.0001$), where we define the null-hypothesis as the MAE for *MS* being larger than or equal to the MAE for *BeyMS*. Results marked with *** indicate that the null-hypothesis was rejected for every fold. Thus, all algorithms (including NMF) provide a significantly larger error for *BeyMS* than for *MS*. In other words, recommendation quality is significantly better for users with mainstream taste than for users who prefer beyond-mainstream music across all recommendation approaches.

These initial results underpin the need to study the characteristics of the *BeyMS* user group that receives worse recommendations. The corresponding experiments are presented in the next section.

4 Characteristics of beyond-mainstream music and listeners

We identify the types of beyond-mainstream music using unsupervised clustering and characterize these types with respect to acoustic features and music genres. Besides, we detect subgroups of beyond-mainstream music listeners by assigning users to these clusters and evaluate the recommendation quality obtained for these subgroups. Finally, we discuss the recommendation quality with respect to openness and diversity. For this, we relate to the definitions given by [9]:

Openness is the across-groups diversity (or categorical diversity) and describes if users of one group also listen to the music of other groups.

Diversity is the within-groups diversity (or thematic diversity) and describes the dissimilarity of music listened to by users within groups.

Based on the findings of [9], we would expect that subgroups with high openness should receive more accurate recommendations than subgroups with high diversity.

4.1 Clustering and characterizing beyond-mainstream music

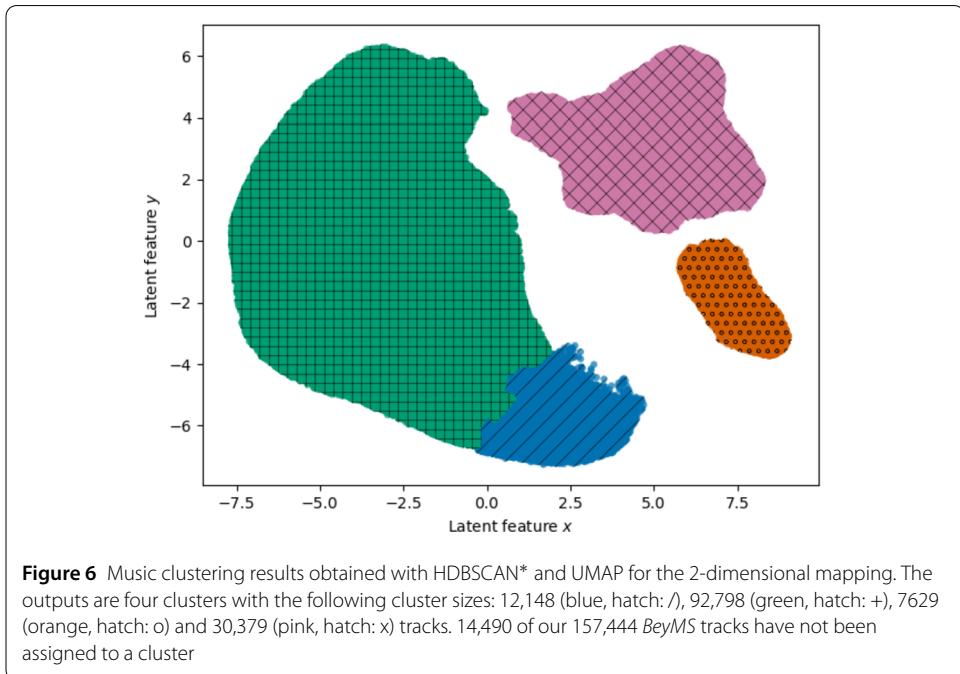
To study the different types of music listened to by the users in our *BeyMS* group, we conduct a cluster analysis. Specifically, we cluster the 157,444 tracks listened to by *BeyMS* users, where each track is described by the eight acoustic features danceability, energy, speechiness, acousticness, instrumentalness, tempo, valence, and liveness (see Sect. 3.1). We scale the value ranges of these features to [0, 1] using min-max normalization. The use of latent representations of musical elements such as tracks was shown to be efficient in the area of music information retrieval [30, 60, 61]. Furthermore, for visually analyzing the obtained music clusters and decreasing computation time, we favor a reduction of dimensionality to two dimensions.

We conduct experiments with a broad body of dimensionality reduction methods, i.e., linear and nonlinear principal component analysis (PCA) [62], locally linear embedding

[63], multidimensional scaling [64], Isomap [65], spectral embedding [66], t-distributed stochastic neighbor embedding (t-SNE) [67] and uniform manifold approximation and projection (UMAP) [68]. We visually inspected the 2-dimensional feature spaces created by these methods with regards to the clustering quality, and we obtained the visually most homogeneous results with UMAP. Moreover, UMAP has already been successfully used in the music domain [30] and thus, we use it for the remainder of our experiments. Specifically, we utilize the open-source implementation of UMAP [69], which requires four parameters: (i) the distance metric M in the input space, (ii) the number of latent dimensions D , (iii) the minimum distance of points in the latent space d_{\min} , and (iv) the number of neighbors of a point N . Based on experimentation and related literature (e.g., [69]), we set the distance metric M to the Euclidean distance, the number of latent dimensions D to 2, the distance d_{\min} to 0.1 and the number of neighbors N to 15.

In a next step, we perform clustering on the dimensionality-reduced acoustic features of tracks. Again, we conduct experiments with various clustering methods, i.e., DBSCAN [70], K-Means [71], Gaussian mixture models [72], affinity propagation [73], spectral clustering [74], hierarchical agglomerative clustering [75], OPTICS [76] and HDBSCAN* [77]. Here, we obtain the best results with respect to cluster cohesion and separation using HDBSCAN*. Furthermore, HDBSCAN* was also already used by related work to cluster music items [78]. We employ the open-source implementation of HDBSCAN* [79] that requires four parameters: (i) the minimum cluster size s_{\min} that defines the minimum size of a group of points to consider a cluster, (ii) the minimum number of samples in the neighborhood of a core point N_{\min} , which quantifies how conservative the clustering is, (iii) ε , which enables the recovery of DBSCAN clusters if the s_{\min} value is not reached, and (iv) the scaling of the distance α , which is another measure of the clustering's conservativeness. In detail, α scales the distance between two points, which determines whether these points are merged into a cluster. This scaling is used in the construction of HDBSCAN*'s hierarchy of clusterings. Again, we find the best-suited parameters based on experimentation and related literature (e.g., [77]). Specifically, we require each cluster to comprise a sufficiently large number of tracks to increase the level of significance of our subsequent experiments. We expect the existence of very small music clusters and thus, search for the optimal value of the minimal cluster size s_{\min} in the search space of {1000; 1025; ...; 1475; 1500}, where we obtain the best results with respect to the within-cluster variance for $s_{\min} = 1375$. Furthermore, tightly packed clusters without any contribution of noise should be favored. In other words, all points within a cluster should be within the neighborhood of at least one core point. Thus, we set the minimal number of samples in the neighborhood $N_{\min} = s_{\min} = 1375$. The remaining two parameters are set to their default values, i.e., $\varepsilon = 0$ and $\alpha = 1$.

Figure 6 shows the results of the clustering process using HDBSCAN* and UMAP for the 2-dimensional mapping. This process leads to four music clusters. Here, the green cluster (hatch: +) is the largest one with 92,798 tracks, followed by the pink cluster (hatch: x) with 30,379 tracks and the blue cluster (hatch: /) with 12,148 tracks. The smallest cluster is the orange one (hatch: o) as it contains 7629 tracks. The remaining 14,490 of our 157,444 *BeyMS* tracks have not been assigned to a cluster and thus, will not be included in further analyses and interpretations. Next, we describe how we name these clusters based on their music genre distributions.



4.1.1 Genre distributions

In Fig. 7, we illustrate the top-10 genres of the four music clusters. For this, we refer to the genre IDF-scores presented in Sect. 3.4 and weight each genre assigned to a track in a cluster with its corresponding IDF-score. For example, if a genre with an IDF-score of 1.4 is assigned to 1000 tracks in a cluster, it is visualized as an aggregated genre IDF-score of 1400 in the corresponding plot of Fig. 7. Based on the genre distributions, we label each cluster according to its top genre.

With respect to the blue cluster (hatch: /) in Plot (a), we find top genres such as “folk” and “singersongwriter”, which typically reflect music with high acousticness. In the remainder of this paper, therefore, we refer to this cluster as C_{folk} . The top genres of the green cluster (hatch: +) in Plot (b) are typical high energy music genres such as “hardrock”, “punk”, “poprock”, and “hiphop”. Based on this, we name this cluster C_{hard} .

For the orange cluster (hatch: o) in Plot (c), we find genres that reflect music with high acousticness and high instrumentalness such as “ambient”, “experimental”, “newage”, and “postrock”. As “ambient” clearly dominates the genre distribution for this cluster, we name this cluster C_{ambi} . Similarly to C_{folk} , this cluster contains music with high acousticness; yet, while C_{folk} is characterized by low instrumentalness music, C_{ambi} is characterized by a high level of instrumentalness. Finally, Plot (d) shows the genre distribution of the pink cluster (hatch: x) with “electronica” as the top genre, which leads to the name C_{elec} for this cluster.

Thus, both, C_{elec} and C_{hard} , consist of high energy music but in contrast to C_{hard} , C_{elec} also comprise high instrumentalness values. This also makes sense when looking at other top genres of C_{elec} such as “deathmetal” and “blackmetal” where guttural vocal techniques are often mistakenly classified as another type of instrument [80].

To compare the genre distributions among the four music clusters, we illustrate the relative genre frequency distribution of the clusters in Fig. 8. The relative frequency of a genre g depicts the fraction of listening events of tracks within a cluster c that are annotated with g . Here, we only show genres with a minimum relative genre frequency of 0.1. We

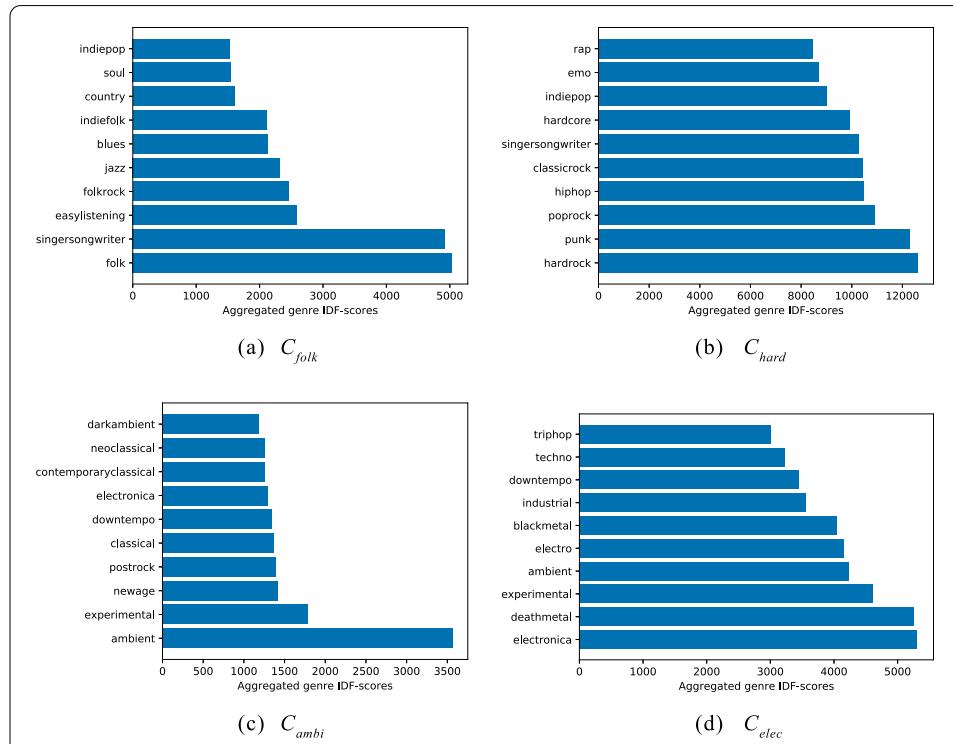


Figure 7 Top-10 genres of the four music clusters C_1 – C_4 according to the aggregated genre IDF-scores. We name the clusters according to the top genre, i.e., (a) blue (hatch: /) → C_{folk} ("folk"), (b) green (hatch: +) → C_{hard} ("hardrock"), (c) orange (hatch: o) → C_{ambi} ("ambient"), and (d) pink (hatch: x) → C_{elec} ("electronica")

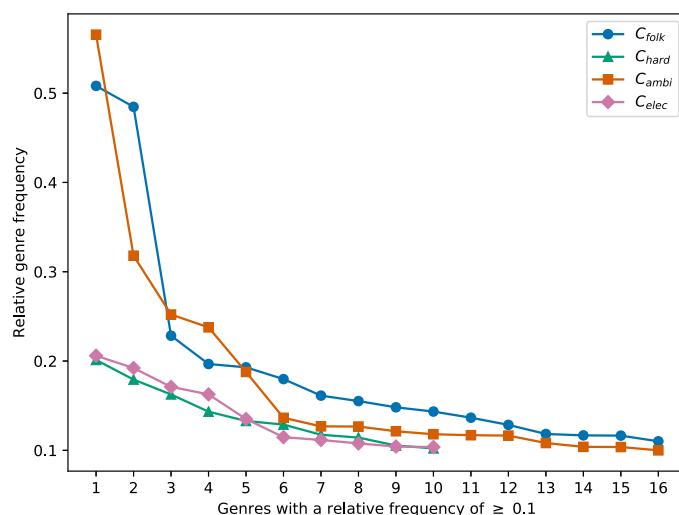


Figure 8 Relative genre frequency distribution of the four music clusters. While there are dominating genres in C_{folk} and C_{ambi} , the genre distribution is more diverse in C_{hard} and C_{elec}

see that there are clearly dominating genres in C_{folk} and C_{ambi} , whereas the genre distributions in C_{hard} and C_{elec} are more evenly distributed. When relating this finding to the findings of Fig. 7, we clearly see that the results correspond to each other: C_{hard} and C_{elec} contain a more diverse genre spectrum (e.g., "hardrock" and "hiphop" are both part of

C_{hard} 's top genres) than C_{folk} and C_{ambi} (e.g., in C_{ambi} 's top genres, we find "ambient" and "darkambient").

4.1.2 Acoustic feature distributions

To understand the musical content of these four music clusters, we analyze the acoustic feature distributions of the four music clusters using boxplots in Fig. 9. This visualization does not show any obvious differences with respect to danceability and tempo among the four clusters. For the acoustic features energy, speechiness, acousticness, valence, and liveness, there are similar values for the cluster pairs C_{folk} and C_{ambi} , and C_{hard} and C_{elec} . We observe differences between these two cluster pairs with respect to energy and acousticness. While C_{hard} and C_{elec} provide high energy values and small acousticness values, C_{folk} and C_{ambi} feature small energy values and high acousticness values.

In contrast, for instrumentalness, we see similar values for the cluster pairs C_{folk} and C_{hard} as well as for C_{ambi} and C_{elec} . We observe very high values for C_{ambi} and C_{elec} , and very small values for C_{folk} and C_{hard} . This difference is also visible in Fig. 6 in the form of the gap between C_{folk} and C_{hard} on the left, and C_{ambi} and C_{elec} on the right.

Summing up, in C_{folk} , we find music with low energy, high acousticness, and low instrumentalness; C_{hard} contains music with high energy, low acousticness, and low instrumentalness; in C_{ambi} , we observe music with low energy, high acousticness, and high instru-

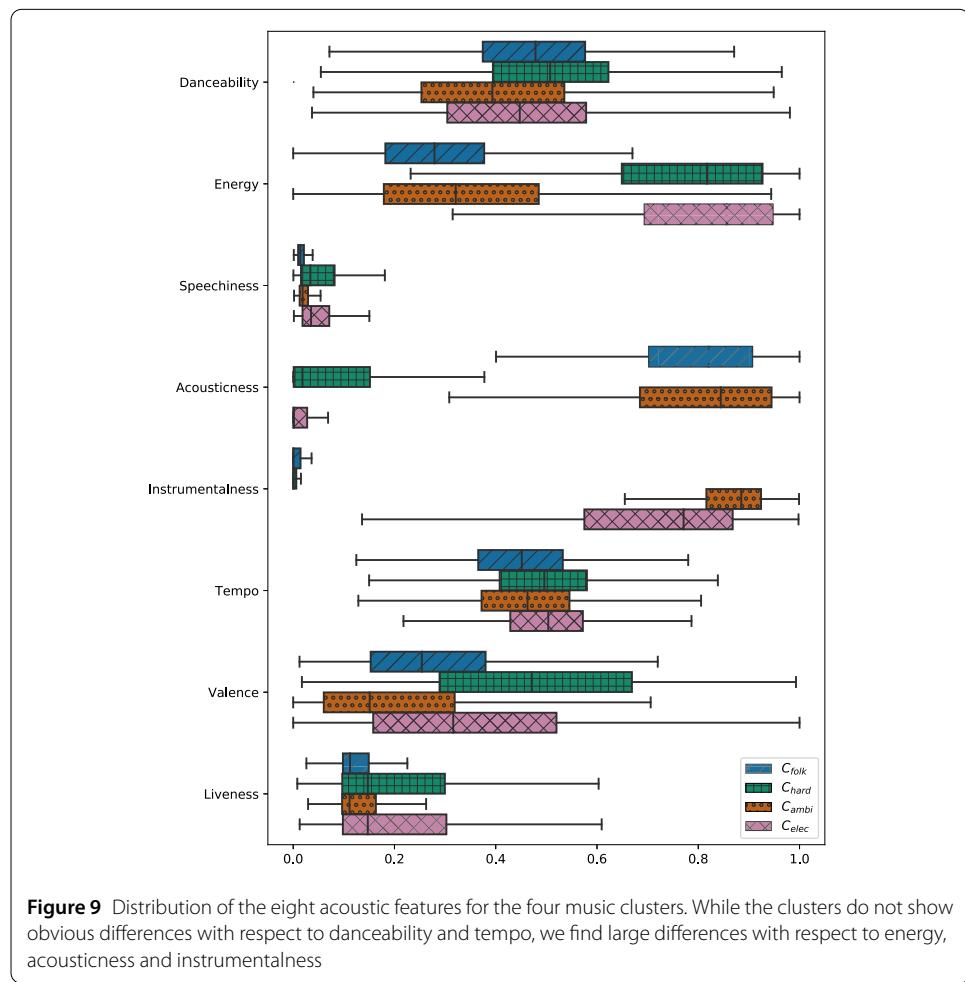


Figure 9 Distribution of the eight acoustic features for the four music clusters. While the clusters do not show obvious differences with respect to danceability and tempo, we find large differences with respect to energy, acousticness and instrumentalness

mentalness; and in C_{elec} , we find high energy, low acousticness, and high instrumentalness. Thus, these findings are in line with the genre distributions presented in Fig. 7.

4.2 Assigning and studying beyond-mainstream music listeners

In the next step, we assign the 2074 *BeyMS* users to the four music clusters to categorize them into four distinct beyond-mainstream subgroups for further analyses.

For each user u , we count the number of listening events $LE_{u,c}$ that u has contributed to the tracks in each cluster c , where $c \in C = \{C_{folk}, C_{hard}, C_{ambi}, C_{elec}\}$. Then, we assign u to the cluster c for which the number of contributed listening events $LE_{u,c}$ is the highest. However, because we have varying cluster sizes, the probability of u listening to a track t of the two larger clusters C_{hard} and C_{elec} is much higher than for the two smaller clusters C_{folk} and C_{ambi} , although C_{folk} and C_{ambi} could be more representative choices for u . Thus, similar to the IDF distribution of genres (see Fig. 5), we take advantage of the IDF scoring to reduce the influence of the larger clusters and to assign higher weights to the smaller clusters. Specifically, these cluster IDF-scores are given by $IDF(c) = \log_{10} \frac{|T|}{|\{t \in T \text{ with } c_t\}|}$, i.e., by relating the number of all tracks $|T|$ to the number of tracks in cluster c where c_t is the music cluster assigned to track t . That lets us define the user–cluster weight $w_{u,c}$ for user u and cluster c as $w_{u,c} = IDF(c) \cdot LE_{u,c}$.

Consequently, users are assigned to the highest weighted music cluster and thus, a subgroup U_c for cluster c is given by $U_c = \{u \in U : \arg \max_{c \in C} (w_{u,c})\}$.

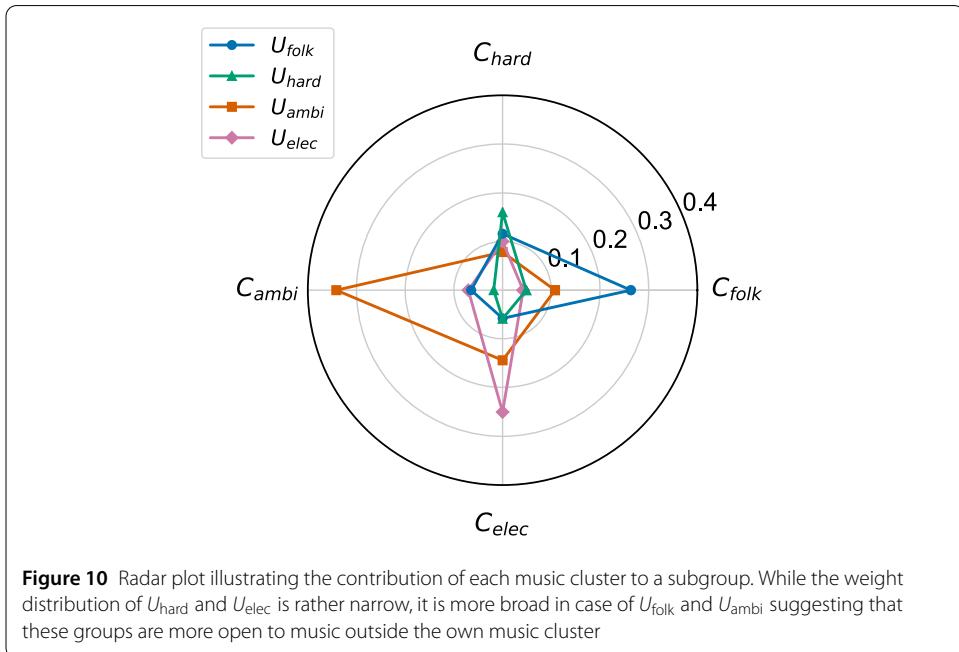
Out of the 2074 *BeyMS* users, we can assign 2073 users to these subgroups. Thus, only 1 user listened to tracks not contained in any cluster in Fig. 6. Similar to the naming scheme of music clusters, we label the subgroups according to the name of their assigned music cluster. Hence, we obtain four subgroups U_{folk} , U_{hard} , U_{ambi} , and U_{elec} .

Table 3 provides basic descriptive statistics of these four resulting subgroups. Here, U_{hard} is the largest subgroup with $|U| = 919$ users, followed by U_{elec} with $|U| = 642$ users, U_{folk} with $|U| = 369$ users, and U_{ambi} with $|U| = 143$ users. The differences with respect to the number of users also correspond to the differences regarding the number of artists $|A|$, the number of tracks $|T|$, and the number of listening events $|LE|$ contained in the clusters. In the case of the number of genres $|G|$, this differs slightly because the users in the smaller U_{ambi} cluster listen to more genres (i.e., 918) than the bigger U_{folk} cluster (i.e., 811). This indicates that the users in U_{ambi} listen to a broader set of music than the users in U_{folk} .

Considering the average number of listening events per user (i.e., $\overline{|LE_u|}$) and the average number of tracks per user (i.e., $\overline{|T_u|}$), we see that, while there is little difference between U_{hard} and U_{elec} with respect to $\overline{|LE_u|}$, $\overline{|T_u|}$ is much higher for U_{elec} (i.e., 670.402) than for U_{hard} (i.e., 557.470). This indicates that, although the number of listening events is nearly the same, users of U_{elec} tend to listen to a wider set of tracks than users of U_{hard} . With

Table 3 Descriptive statistics of the four subgroups. Here, $|U|$ is the number of users, $|A|$ is the number of artists, $|T|$ is the number of tracks, $|LE|$ is the number of listening events, $|G|$ is the number of genres, $\overline{|LE_u|}$ is the average number of listening events per user, $\overline{|T_u|}$ is the average number of tracks per user and \overline{Age} is the average age (along with the standard deviation) of users in the group

Subgroup	$ U $	$ A $	$ T $	$ LE $	$ G $	$\overline{ LE_u }$	$\overline{ T_u }$	\overline{Age} (std.)
U_{folk}	369	9559	72,663	702,635	811	1904.160	549.650	27.599 (± 10.369)
U_{hard}	919	11,966	107,952	2,150,246	1274	2339.767	557.470	23.867 (± 8.912)
U_{ambi}	143	6869	39,649	224,327	918	1568.720	473.308	29.571 (± 14.138)
U_{elec}	642	11,814	105,907	1,416,354	1005	2206.159	670.402	24.639 (± 7.886)



respect to the average age of the users \overline{Age} , we see that the users of U_{folk} and U_{ambi} are the oldest ones, and users of U_{hard} and U_{elec} are the youngest ones. However, it is worth noting that the group with the highest average age (i.e., U_{ambi}) also shows by far the highest standard deviation of age (i.e., 14.138 years).

In Fig. 10, we show the contribution of each music cluster to each subgroup in the form of a radar plot. For this, we use the user-cluster weights $w_{u,c}$ introduced before and calculate the average weight over all users in cluster c . One consequence of the IDF scoring applied to $w_{u,c}$ is that the weight contributions of a user group to the four clusters does not sum up to 1, which eventually influences the interpretation of the values shown in Fig. 10. However, in return, these values account for the varying cluster sizes and can also be interpreted as preference weights for a user group towards a specific music cluster.

We observe that the weight distribution of the two larger subgroups U_{hard} and U_{elec} is rather narrow, which indicates that these users do not listen to many tracks of other clusters. Contrary to that, the weights of the two smaller subgroups U_{folk} and U_{ambi} are more broadly distributed over the four music clusters. This suggests that users of U_{folk} and U_{ambi} are more open to music outside of their own music cluster than users of U_{hard} and U_{elec} .

4.2.1 Correlation of music clusters and beyond-mainstream subgroups

To better understand the correlations and connections between the music clusters and subgroups, we plot the Pearson correlation matrix of the four music clusters as a heatmap in Fig. 11. Here, we represent each music cluster c by a 2073-dimensional vector (i.e., one entry for each user) consisting of the user–cluster weights $w_{u,c}$, introduced before. Each element in the matrix is then calculated using the Pearson correlation measure based on these cluster vectors. For example, if there is a positive correlation between two clusters, we assume that a user who enjoys music from the one cluster likely also enjoys music from the other cluster. This can give us also an indication of the openness of a subgroup

Figure 11 Pearson correlation matrix of the four music clusters. While C_{hard} has solely negative correlations with all other clusters, and thus, listeners of C_{hard} seem to be the most closed subgroup, C_{ambi} has positive correlations with C_{folk} and C_{elec} , and thus, listeners of C_{ambi} seem to be the most open subgroup

	C_{folk}	-0.28	0.14	-0.32
C_{hard}	-0.28	1	-0.66	-0.74
C_{ambi}	0.14	-0.66	1	0.23
C_{elec}	-0.32	-0.74	0.23	1
	C_{folk}	C_{hard}	C_{ambi}	C_{elec}

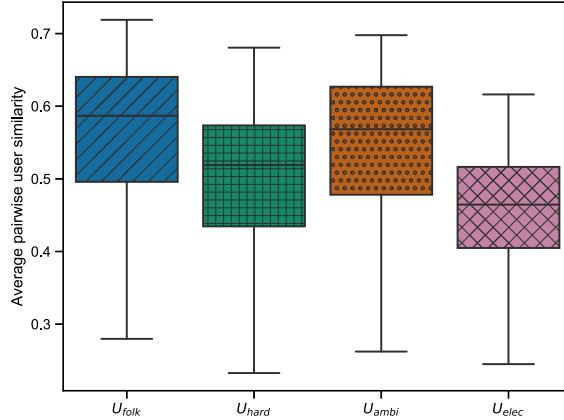


Figure 12 Boxplots showing the average pairwise user similarity of the four subgroups using the cosine similarity calculated on the users' genre distributions. While the users in U_{hard} and U_{elec} exhibit a more diverse listening behavior, users in U_{folk} and U_{ambi} tend to listen to more similar, i.e., less diverse, music genres

for music mainly listened to by other subgroups. Specifically, for C_{folk} , we see a positive correlation between C_{folk} and C_{ambi} , and a negative correlation between C_{folk} and both, C_{hard} as well as C_{elec} . Users listening to the music of C_{hard} seem to represent the most closed subgroup as C_{hard} because it solely has negative correlations with all other clusters, especially with C_{ambi} and C_{elec} . In contrast, users listening to the music of C_{ambi} seem to represent the most open subgroup as C_{ambi} has positive correlations with two other clusters, i.e., C_{folk} and C_{elec} . The fourth cluster, C_{elec} , is negatively correlated with C_{folk} and especially with C_{hard} , and positively correlated with C_{ambi} . These results are also in line with the ones shown in Fig. 10, in which we identify the users of U_{ambi} as more open music listeners than the ones of U_{hard} .

In order to relate the openness of the subgroups to the diversity of the users within the subgroups, we calculate the average pairwise user similarity using the cosine similarity metric computed on the users' genre distributions, i.e., number of listening events per genre. Figure 12 shows the resulting boxplots for the four identified subgroups (i.e., C_{folk} , C_{hard} , C_{ambi} , and C_{elec}). Figure 12 shows that users in U_{hard} and U_{elec} have a rather small average pairwise user similarity and, thus, exhibit a more diverse listening behavior, whereas users in U_{folk} and U_{ambi} tend to listen to more similar music genres and, thus, have a narrow listening behavior within the group. Summed up, we find pronounced differences with respect to openness and diversity across the subgroups. Although U_{ambi} is the most open

subgroup (i.e., also listens to music of other subgroups), it is also the least diverse subgroup (i.e., the users within the group listen to very similar music). That observation is in line with what is shown in Figs. 7, and Fig. 8. Here, we see that C_{ambi} , i.e., the most tightly connected music cluster to U_{ambi} , contains the dominating genre “ambient” as well as genres that are strongly associated with this dominating genre (e.g., “darkambient”). For U_{hard} , we observe the opposite. While it is the least open subgroup, it is also the most diverse one (e.g., it contains “hardrock” as well as “hiphop” listeners).

4.2.2 Recommendations for beyond-mainstream user subgroups

In Sect. 3.5, we have shown that the recommendation accuracy of four personalized recommendation algorithms is significantly worse for *BeyMS* users than for *MS* users. Now, we extend this analysis and evaluate the recommendation accuracy of these algorithms for the four subgroups (i.e., U_{folk} , U_{hard} , U_{ambi} , and U_{elec}).

Table 4 shows our results with respect to the mean absolute error (MAE). Additionally, we analyze these results with respect to statistically significant differences in Table 5 by performing ANOVA ($\alpha = 0.01$) and a subsequent Tukey-HSD test ($\alpha = 0.05$). Here, we report pairwise differences as significant (marked with **), if both ANOVA and Tukey-HSD were significant across all five folds (see Sect. 3.5 for details on the experimental setup).

We see that among all algorithms, the significantly worst accuracy results (i.e., the highest MAE scores) are achieved for the U_{hard} subgroup. Next, U_{folk} , U_{ambi} and U_{elec} reach significantly better (i.e., lower MAE scores) than U_{hard} for all algorithms. However, there is no statistically significant difference between the recommendation accuracy of U_{folk} and U_{elec} . The overall best accuracy results (i.e., lowest MAE scores) are reached for the U_{ambi} subgroup. These results are also statistically significant when compared with the other subgroups for the NMF algorithm. NMF also gives the overall best accuracy results for

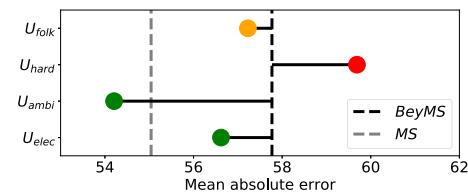
Table 4 Mean absolute error (MAE) measurements for the four subgroups and four personalized recommendation algorithms. NMF (in bold) outperforms all other algorithms for all subgroups. Among the subgroups, the best accuracy results (i.e., lowest MAE scores) are reached by U_{ambi} , while the worst accuracy results (i.e., highest MAE scores) are reached by U_{hard} . To facilitate comparison, we also show the MAE measurements for the *BeyMS* and *MS* user groups

Subgroup	UserItemAvg	UserKNN	UserKNNAvg	NMF
U_{folk}	63.2143	70.3049	67.4406	57.2278
U_{hard}	65.1464	73.1949	69.2855	59.6887
U_{ambi}	60.5558	69.8315	65.5708	54.2073
U_{elec}	62.2894	71.0387	66.1499	56.6209
<i>BeyMS</i>	63.4608	71.6694	67.5856	57.7703
<i>MS</i>	61.2562	68.4894	63.3985	54.8182

Table 5 Statistically significant differences between pairs of subgroups, as determined by ANOVA ($\alpha = 0.01$) and a subsequent Tukey-HSD test ($\alpha = 0.05$)

Subgroup	UserItemAvg				UserKNN				UserKNNAvg				NMF			
	U_{folk}	U_{hard}	U_{ambi}	U_{elec}												
U_{folk}	**	**			**				**				**	**	**	**
U_{hard}	**	**	**	**	**	**	**	**	**	**	**	**	**	**	**	**
U_{ambi}	**	**			**				**				**	**		**
U_{elec}	**				**				**				**	**		**

Figure 13 Comparison of the mean absolute error (MAE) scores reached by NMF for the four subgroups with the ones reached by NMF for *BeyMS* (black dashed line) and *MS* (gray dashed line). While specific subgroups (i.e., U_{hard}) are treated in an unfair way by recommendation algorithms, others (i.e., U_{ambi}) are not



all subgroups, which is in line with our results presented in Sect. 3.5 and in our previous work [4].

Furthermore, we find a relationship between openness, diversity, and recommendation quality. Here, U_{hard} is the least open but most diverse subgroup and gets the worst recommendations, while U_{ambi} is the most open but least diverse subgroup and gets the best recommendations. This is in line with the findings of [9], who have shown that users are more likely to accept recommendations from different groups (i.e., openness) rather than varied within a group (i.e., diversity). Thus, we find a relationship between the quality of recommendations provided to beyond-mainstream music listeners and openness as well as diversity patterns of these users.

Finally, in Fig. 13, we visually compare the MAE scores reached by the best performing approach NMF for the four subgroups. Additionally, we depict the MAE score for *BeyMS* as a black dashed line and the MAE score for *MS* as a gray dashed line. We see that U_{hard} reaches worse results than *BeyMS* while U_{folk} and U_{elec} reach slightly better results than *BeyMS*. Interestingly, U_{ambi} not only reaches better results than *BeyMS* but also better results than *MS*. Although this improvement over *MS* is not statistically significant (according to a one-tailed Mann–Whitney-U test with $\alpha = 0.0001$), it shows that there is a large variety among *BeyMS* users, where specific subgroups (i.e., U_{hard}) are disadvantaged in terms of recommendation accuracy by recommendation algorithms while others (i.e., U_{ambi}) are not.

5 Conclusions and future work

In this paper, we shed light on the characteristics of beyond-mainstream music and music listeners. As our first contribution, we identified 2074 beyond-mainstream music listeners (i.e., *BeyMS*) in the Last.fm platform, and subsequently created a novel dataset called *LFM-BeyMS* based on the listening histories of these users. We further enriched this dataset with (i) acoustic features of music tracks gathered from Spotify, and (ii) genre information of tracks derived from Last.fm tags and matched with the Spotify microgenre taxonomy. Additionally, for reasons of comparability, *LFM-BeyMS* contains data of 2074 Last.fm users listening to mainstream music. Using this dataset, as our second contribution, we validated related research by showing that beyond-mainstream music listeners receive a significantly lower recommendation accuracy than mainstream music listeners by four standard recommendation algorithms (i.e., UserItemAvg, UserKNN, UserKNNAvg and NMF).

As our third contribution, we applied the clustering algorithm HDBSCAN* on the acoustic features of tracks listened by *BeyMS* and identified four clusters of beyond-mainstream music: (i) C_{folk} , music with high acousticness such as “folk”, (ii) C_{hard} , high energy music such as “hardrock”, (iii) C_{ambi} , music with high acousticness and instrumen-

talness such as “ambient”, and (iv) C_{elec} , music with high energy and instrumentality such as “electronica”.

As our fourth contribution, we mapped these clusters to our *BeyMS* users, which led to four beyond-mainstream subgroups: (i) U_{folk} , (ii) U_{hard} , (iii) U_{ambi} , and (iv) U_{elec} . We analyzed these subgroups with respect to their openness (i.e., across-groups diversity—do users of one group listen to music of other groups?) and diversity (i.e., within-groups diversity—how dissimilar is the music listened to by users within groups?). Here, we found large differences between U_{hard} and U_{ambi} . Although U_{hard} is the most closed subgroup (i.e., users do not listen to music of other subgroups), it is also the most diverse subgroup (i.e., users listen to a diverse set of genres such as “hardrock” and “hiphop”). For U_{ambi} , we get opposite results: while it is the most open subgroup (i.e., users listen to music of other subgroups as well), it is also the least diverse one (i.e., the users within the group listen to very similar music such as “ambient” and “darkambient”). We related these characteristics of the subgroups to the recommendation quality of the four recommendation algorithms UserItemAvg, UserKNN, UserKNNAvg and NMF. Here, we found that U_{hard} got music recommendations with lowest accuracy, while U_{ambi} got music recommendations with highest accuracy. This is in line with related research [9], which has shown that openness is stronger correlated with accurate recommendations than diversity. U_{ambi} even received better recommendations than the group of mainstream music listeners. This result highlights that there are large differences between the subgroups of beyond-music listeners. Finally, to foster reproducibility of our research, we provide our novel *LFM-BeyMS* dataset via Zenodo as well as our source code via Github.

We believe that our findings provide useful insights for creating user models and recommendation algorithms that better serve beyond-mainstream music listeners. As it was shown in [4], beyond-mainstream music listeners tend to have larger user profile sizes than users interested in mainstream music, which means that they provide a substantial amount of listening interaction data for services such as Last.fm and Spotify. We assume that improving the recommendation quality for this active user group also leads to another effect, namely a more prominent exposure of (long-tail) music artists due to a better-connected recommendation network [81]. We leave such investigations to future work.

Limitations Despite the merits of this work, we are aware of its limitations. The first limitation we recognize is that our analyses are based on a sample of the Last.fm community. The extent to which their listening behavior is representative of the Last.fm community at large, or similar music streaming communities such as Spotify, needs further investigation.

Next, since we conducted a comparative study of the accuracy of recommender systems algorithms—and were therefore not interested to beat state-of-the-art algorithms—we focused on traditional algorithms (e.g., KNN-based collaborative filtering) instead of investigating the most current deep learning architectures, which would also require a much higher computational effort. Furthermore, an award-winning-paper by Dacrema et al. [82] has recently shown that traditional algorithms are able to outperform almost all deep learning architectures.

Future work While our work serves as a first milestone towards better characterizing beyond-mainstream music and listeners of such music, future work should focus on user modeling techniques to individually target the different subgroups, for example by integrating knowledge about openness and diversity. With respect to analyzing openness and

diversity of users and user groups, we would also like to work on a more formal definition of these dimensions, which would not only allow us to measure them more precisely but also to integrate them into the recommendation calculation process.

Additionally, since previous research has shown that the listener's cultural background impacts the quality of music recommendations [48], we plan to compare the cultural and socioeconomic aspects of beyond-mainstream and mainstream music listeners. We plan to employ these aspects by means of Hofstede's cultural dimensions [83] and the World Happiness Report [84].

Finally, another avenue for future work is the research in the area of fair music recommender systems. Here, we plan to build user models that are capable of accounting for the complex characteristics of beyond-mainstream music listeners presented in this paper. While we believe that more specialized user models could help to provide better recommendations for users who currently receive worse recommendations (e.g., the U_{hard} subgroup identified in this paper), we also aim to highlight that such user models still need to be generalizable to avoid any unfair treatment of other users. Hence, future research should work on achieving a specialization-generalization trade-off in music recommender systems. We hope that our open *LFM-BeyMS* dataset as well as our source code will be of use to the scientific community for subsequent analyses.

Acknowledgements

This work is supported by the Know-Center GmbH within the COMET—Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (bmvt), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG. program.

Funding

This work is funded by the TU Graz Open Access Publishing Fund and the Austrian Science Fund (FWF): V579.

Availability of data and materials

The *CultMRS* dataset utilized during the current study is made available on Zenodo, accessible via the following DOI: <https://doi.org/10.5281/zenodo.3477842> and further described in [50]. Additionally, we provide the novel *LFM-BeyMS* dataset on Zenodo: <https://doi.org/10.5281/zenodo.3784764>. The entire Python-based implementation of the experiments conducted in this study is publicly available at <https://github.com/pmuellner/supporttheunderground>.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors contributed to manuscript revision, read, and approved the submitted version.

Author details

¹Know-Center GmbH, Graz, Austria. ²University of Innsbruck, Innsbruck, Austria. ³Utrecht University, Utrecht, The Netherlands. ⁴Johannes Kepler University Linz, Linz, Austria. ⁵Linz Institute of Technology AI Lab, Linz, Austria. ⁶Graz University of Technology, Graz, Austria.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 7 August 2020 Accepted: 23 February 2021 Published online: 30 March 2021

References

1. Schedl M, Knees P, McFee B, Bogdanov D, Kaminskas M (2015) Music recommender systems. In: Recommender systems handbook, pp 453–492
2. Abdollahpour H, Mansouri M, Burke R, Mobasher B (2019) The unfairness of popularity bias in recommendation. In: RMSE workshop held in conjunction with the 13th ACM conference on recommender systems (RecSys)
3. Celma O (2009) Music recommendation and discovery in the long tail. PhD thesis, Universitat Pompeu Fabra
4. Kowald D, Schedl M, Lex E (2020) The unfairness of popularity bias in music recommendation: a reproducibility study. In: European conference on information retrieval. Springer, Berlin, pp 35–42
5. Celma O, Cano P (2008) From hits to niches?: or how popular artists can bias music recommendation and discovery. In: Proceedings of KDD '08 (Netflix prize workshop)

6. Celma O (2010) Music recommendation and discovery—the long tail, long tail, and long play in the digital music space. Springer
7. van den Oord A, Dieleman S, Schrauwen B (2013) Deep content-based music recommendation. In: Proceedings of NIPS '2013. Curran Associates, Red Hook, pp 2643–2651
8. Goel S, Broder A, Gabrilovich E, Pang B (2010) Anatomy of the long tail: ordinary people with extraordinary tastes. In: Proceedings of the third ACM international conference on web search and data mining, pp 201–210
9. Tintarev N, Dennis M, Masthoff J (2013) Adapting recommendation diversity to openness to experience: a study of human behaviour. In: Carberry S, Weibelzahl S, Micarelli A, Semeraro G (eds) User modeling, adaptation, and personalization. Springer, Berlin, pp 190–202
10. Luo X, Zhou M, Xia Y, Zhu Q (2014) An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Trans Ind Inform* 10(2):1273–1284
11. Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. *ACM Trans Inf Syst* 22(1):5–53
12. Schedl M, Zamani H, Chen C-W, Deldjoo Y, Elahi M (2018) Current challenges and visions in music recommender systems research. *Int J Multimed Inf Retr* 7(2):95–116
13. Haas R, Brandes V (2010) Music that works: contributions of biology, neurophysiology, psychology, sociology, medicine and musicology. Springer Science & Business Media
14. Adorno TW (1988) Introduction to the sociology of music. Burns & Oates
15. Deutsch D (2013) Psychology of music. Elsevier
16. Laplante A (2014) Improving music recommender systems: what can we learn from research on music tastes? In: Proceedings of the International Society for Music Information Retrieval conference (ISMIR)
17. Rentfrow PJ, Gosling SD (2007) The content and validity of music-genre stereotypes among college students. *Psychol Music* 35(2):306–326
18. Kim Y, Aiello LM, Quercia D (2020) Pepmusic: motivational qualities of songs for daily activities. *EPJ Data Sci* 9(1):13
19. Juslin PN, Sloboda JA (2001) Music and emotion: theory and research. Oxford University Press
20. Zentner M, Grandjean D, Scherer KR (2008) Emotions evoked by the sound of music: characterization, classification, and measurement. *Emotion* 8(4):494
21. Juslin PN, Laukka P (2004) Expression, perception, and induction of musical emotions: a review and a questionnaire study of everyday listening. *J New Music Res* 33(3):217–238
22. Yang Y-H, Chen HH (2011) Music emotion recognition. CRC Press
23. Ferwerda B, Schedl M, Tkalcic M (2015) Personality & emotional states: understanding users' music listening needs. In: Late-breaking results of 23rd international conference on user modeling, adaptation and personalization (UMAP)
24. Goldberg LR (1993) The structure of phenotypic personality traits. *Am Psychol* 48(1):26
25. Schubert E (2007) The influence of emotion, locus of emotion and familiarity upon preference in music. *Psychol Music* 35(3):499–515
26. Pereira CS, Teixeira J, Figueiredo P, Xavier J, Castro SL, Brattico E (2011) Music and emotions in the brain: familiarity matters. *PLoS ONE* 6(11):e27241
27. Moore JL, Chen S, Turnbull D, Joachims T (2013) Taste over time: the temporal dynamics of user preferences. In: Proceedings of the International Society for Music Information Retrieval conference (ISMIR), pp 401–406
28. Barone MD, Bansal J, Woolhouse MH (2017) Acoustic features influence musical choices across multiple genres. *Front Psychol* 8:931
29. Gong B, Kaya M, Tintarev N (2020) Contextual personalized re-ranking of music recommendations through audio features. Master's thesis, TU Delft
30. Zangerle E, Pichl M (2018) Content-based user models: modeling the many faces of musical preference. In: Proceedings of the 19th International Society for Music Information Retrieval conference 2018 (ISMIR 2018), pp 709–716
31. Ekstrand MD, Tian M, Azpiazu IM, Ekstrand JD, Anuyah O, McNeill D, Pera MS (2018) All the cool kids, how do they fit in?: popularity and demographic biases in recommender evaluation and effectiveness. In: Conference on fairness, accountability and transparency, pp 172–186
32. Brynjolfsson E, Hu YJ, Smith MD (2006) From niches to riches: anatomy of the long tail. *Sloan Manag Rev* 47(4):67–71
33. Jannach D, Lerche L, Kamehkhosh I, Jugovac M (2015) What recommenders recommend: an analysis of recommendation biases and possible countermeasures. *User Model User-Adapt Interact* 25(5):427–491
34. Harper FM, Konstan JA (2015) The movielens datasets: history and context. *ACM Trans Interact Intell Syst* 5(4):1–19
35. Cheng R, Tang B (2016) A music recommendation system based on acoustic features and user personalities. In: Pacific-Asia conference on knowledge discovery and data mining. Springer, Berlin, pp 203–213
36. Kaminskas M, Ricci F, Schedl M (2013) Location-aware music recommendation using auto-tagging and hybrid matching. In: Proceedings of RecSys '2013. ACM, Hong Kong, pp 17–24
37. Donaldson J (2007) A hybrid social-acoustic recommendation system for popular music. In: Proceedings of RecSys '2007. ACM, New York, pp 187–190
38. Aggarwal CC (2016) Ensemble-based and hybrid recommender systems. In: Recommender systems, pp 199–224
39. Zangerle E, Pichl M (2018) Content-based user models: modeling the many faces of musical preference. In: 19th International Society for Music Information Retrieval conference (ISMIR)
40. Lee K, Lee K (2011) My head is your tail: applying link analysis on long-tailed music listening behavior for music recommendation. In: Proceedings of the 5th ACM conference on recommender systems, pp 213–220
41. Lex E, Kowald D, Schedl M (2020) Modeling popularity and temporal drift of music genre preferences. *Trans Int Soc Music Inf Retr* 3(1):17–30
42. Kowald D, Lex E, Schedl M (2019) Modeling artist preferences of users with different music consumption patterns for fair music recommendations. In: Late-breaking-results of the 20th annual conference of the International Society for Music Information Retrieval (ISMIR)
43. Kowald D, Kopeinik S, Lex E (2017) The tagrec framework as a toolkit for the development of tag-based recommender systems. In: Adjunct publication of the 25th conference on user modeling, adaptation and personalization, pp 23–28
44. Bauer C (2019) Allowing for equal opportunities for artists in music recommendation. In: 1st workshop on Designing Human-Centric Music Information Research systems in conjunction with ISMIR conference

45. Pichl M, Zangerle E, Specht G (2016) Understanding playlist creation on music streaming platforms. In: IEEE international symposium on multimedia, ISM 2016, pp 475–480
46. Andersen JS (2014) Using the echo nest's automatically extracted music features for a musical purpose. In: 4th international workshop on cognitive information processing (CIP), pp 1–6
47. McVicar M, Freeman T, De Bie T (2011) Mining the correlation between lyrical and audio features and the emergence of mood. In: Proceedings of the 11th International Society for Music Information Retrieval conference (ISMIR), pp 783–788
48. Zangerle E, Pichl M, Schedl M (2020) User models for culture-aware music recommendation: fusing acoustic and cultural cues. *Trans Int Soc Music Inf Retr* 3(1):1–16. <https://doi.org/10.5334/tismir.37>
49. Schedl M (2016) The lfm-1b dataset for music retrieval and recommendation. In: Proceedings of the 2016 ACM on international conference on multimedia retrieval. ACM, New York, pp 103–110
50. Zangerle E Culture-aware music recommendation dataset. <https://doi.org/10.5281/zenodo.3477842>
51. Bauer C, Schedl M (2019) Global and country-specific mainstreamness measures: definitions, analysis, and usage for improving personalized music recommendation systems. *PLoS ONE* 14(6):e0217389. <https://doi.org/10.1371/journal.pone.0217389>
52. Kendall MG (1938) A new measure of rank correlation. *Biometrika* 30(1/2):81–93
53. Sheather SJ (2004) Density estimation. *Stat Sci* 19(4):588–597
54. Davis RA, Lii K-S, Politis DN (2011) Remarks on some nonparametric estimates of a density function. In: Selected works of Murray Rosenblatt. Springer, New York, pp 95–100
55. Quarteroni A, Sacco R, Saleri F (2007) Numerical mathematics. Springer, New York
56. Jones KS (1972) A statistical interpretation of term specificity and its application in retrieval. *J Doc* 28(1):11–21
57. Schedl M, Bauer C (2017) Distance-and rank-based music mainstreamness measurement. In: Adjunct publication of the 25th conference on user modeling, adaptation and personalization. ACM, New York, pp 364–367
58. Koren Y (2010) Factor in the neighbors: scalable and accurate collaborative filtering. *ACM Trans Knowl Discov Data* 4(1):1
59. Willmott CJ, Matsuura K (2005) Advantages of the mean absolute error (mae) over the root mean square error (RMSE) in assessing average model performance. *Clim Res* 30(1):79–82
60. Moore JL, Chen S, Joachims T, Turnbull D (2012) Learning to embed songs and tags for playlist prediction. In: Proceedings of the 12th International Society for Music Information Retrieval conference (ISMIR), vol 12, pp 349–354
61. Levy M, Sandler M (2008) Learning latent semantic models for music from social tags. *J New Music Res* 37(2):137–150
62. Tipping ME, Bishop CM (1999) Mixtures of probabilistic principal component analyzers. *Neural Comput* 11(2):443–482
63. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(5500):2323–2326
64. Kruskal JB (1964) Nonmetric multidimensional scaling: a numerical method. *Psychometrika* 29(2):115–129
65. Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. *Science* 290(5500):2319–2323
66. Ng AY, Jordan MI, Weiss Y (2002) On spectral clustering: analysis and an algorithm. In: Advances in neural information processing systems, pp 849–856
67. van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. *J Mach Learn Res* 9:2579–2605
68. McInnes L, Healy J, Melville J (2018) UMAP: uniform manifold approximation and projection for dimension reduction. *J Open Sour Softw*
69. McInnes L, Healy J, Saul N, Grossberger L (2018) UMAP: uniform manifold approximation and projection. *J Open Sour Softw* 3(29):861
70. Ester M, Kriegel H-P, Sander J, Xu X et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD, vol 96, pp 226–231
71. Bishop CM (2006) Pattern recognition and machine learning Springer, New York, pp 424–429
72. Reynolds D (2015) Gaussian mixture models. In: Encyclopedia of biometrics, pp 827–832
73. Frey BJ, Dueck D (2007) Clustering by passing messages between data points. *Science* 315(5814):972–976
74. Shi J, Malik J (2000) Normalized cuts and image segmentation. *Departmental Papers (CIS)*, 107
75. Murtagh F, Legendre P (2014) Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion? *J Classif* 31(3):274–295
76. Ankerst M, Breunig MM, Kriegel H-P, Sander J (1999) Optics: ordering points to identify the clustering structure. In: ACM sigmod record, vol 28. ACM, New York, pp 49–60
77. McInnes L, Healy J (2017) Accelerated hierarchical density based clustering. In: Data mining workshops (ICDMW), 2017 IEEE international conference on. IEEE, New York, pp 33–42
78. Yoo S, Lee K (2017) A data-driven approach to identifying music listener groups based on users' playrate distributions of listening events. In: Adjunct publication of the 25th conference on user modeling, adaptation and personalization, pp 77–81
79. McInnes L, Healy J, Astels S (2017) hdbSCAN: hierarchical density based clustering. *J Open Sour Softw* 2(11):205
80. York W (2004) Voices from hell—the dark, not-so-dulcet cookie monster vocals of extreme metal. *The San Francisco Bay Guardian*, 14–20
81. Lamprecht D, Strohmaier M, Helic D (2017) A method for evaluating discoverability and navigability of recommendation algorithms. *Comput Soc Netw* 4(1):9
82. Dacrema MF, Cremonesi P, Jannach D (2019) Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In: Proceedings of the 13th ACM conference on recommender systems, RecSys 2019, Copenhagen, Denmark, September 16–20, 2019, pp 101–109
83. Hofstede G, Hofstede GJ, Minkov M (2010) Cultures and organizations: software of the mind, 3rd edn. McGraw-Hill, New York
84. Helliwell JF, Layard R, Sachs J (2016) World happiness report 2016 update. Sustainable Development Solutions Network, New York

P13 Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected? (2021)

Fairness and Popularity Bias in Recommender Systems

[P13] Lesota, O., Melchiorre, A., Rekabsaz, N., Brandl, S., **Kowald, D.**, Lex, E., Schedl, M. (2021). Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected? In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'2021)*, pp. 601-606.

DOI: <https://doi.org/10.1145/3460231.3478843>



Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected?

Oleg Lesota

oleg.lesota@jku.at

Johannes Kepler University Linz and
Linz Institute of Technology
Austria

Stefan Brandl

stefan.brandl@jku.at

Johannes Kepler University Linz
Austria

Alessandro B. Melchiorre

alessandro.melchiorre@jku.at

Linz Institute of Technology
Austria

Navid Rekabsaz

navid.rekabsaz@jku.at

Johannes Kepler University Linz and
Linz Institute of Technology
Austria

Elisabeth Lex

elisabeth.lex@tugraz.at

Graz University of Technology
Austria

Dominik Kowald

dkowald@know-center.at

Know-Center GmbH
Austria

Markus Schedl*

markus.schedl@jku.at

Johannes Kepler University Linz and
Linz Institute of Technology
Austria

ABSTRACT

Several studies have identified discrepancies between the popularity of items in user profiles and the corresponding recommendation lists. Such behavior, which concerns a variety of recommendation algorithms, is referred to as popularity bias. Existing work predominantly adopts simple statistical measures, such as the difference of mean or median popularity, to quantify popularity bias. Moreover, it does so irrespective of user characteristics other than the inclination to popular content. In this work, in contrast, we propose to investigate popularity differences (between the user profile and recommendation list) in terms of median, a variety of statistical moments, as well as similarity measures that consider the entire popularity distributions (Kullback-Leibler divergence and Kendall's τ rank-order correlation). This results in a more detailed picture of the characteristics of popularity bias. Furthermore, we investigate whether such algorithmic popularity bias affects users of different genders in the same way. We focus on music recommendation and conduct experiments on the recently released standardized LFM-2b dataset, containing listening profiles of Last.fm users. We investigate the algorithmic popularity bias of seven common recommendation algorithms (five collaborative filtering and two baselines). Our experiments show that (1) the studied metrics provide novel insights into popularity bias in comparison with only using average differences, (2) algorithms less inclined towards popularity bias amplification do not necessarily perform worse in terms of

utility (NDCG), (3) the majority of the investigated recommenders intensify the popularity bias of the female users.

CCS CONCEPTS

- Information systems → Recommender systems.

KEYWORDS

music recommendation, popularity bias, fairness, gender

ACM Reference Format:

Oleg Lesota, Alessandro B. Melchiorre, Navid Rekabsaz, Stefan Brandl, Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2021. Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected?. In *Fifteenth ACM Conference on Recommender Systems (RecSys '21)*, September 27–October 1, 2021, Amsterdam, Netherlands. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3460231.3478843>

1 INTRODUCTION

Popularity bias in recommender systems refers to a disparity of item popularities in the recommendation lists. Most commonly, this means that a disproportionately higher number of popular items than less popular ones are recommended [8]. The existence of such a popularity bias has been evidenced in different domains already, e.g., movies [3], music [12], or product reviews [1]. Collaborative filtering recommenders are particularly prone to popularity biases because the data they are trained on already exhibit an imbalance towards popular items, i.e., more user-item interactions are available for popular items than less popular ones [2].

The distribution of item popularities in most domains, in particular in the music domain, which we target in this work, shows a long-tail characteristic [5]. A recommendation algorithm introduces no further *algorithmic bias* when the distribution of popularity values of recommended items (tracks) exactly matches the distribution of popularity values of already consumed items (listening history) for each user.

*This is the corresponding author.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

RecSys '21, September 27–October 1, 2021, Amsterdam, Netherlands

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8458-2/21/09.

<https://doi.org/10.1145/3460231.3478843>

We identify two shortcomings of existing studies of popularity bias: First, popularity bias is commonly quantified using simple statistical aggregation metrics, predominantly comparing arithmetic means computed on some count of the user-item interactions [3, 12]. These are not robust against outliers often present in music listening data. Second, popularity bias is typically studied irrespective of user characteristics. Therefore, the extent to which users of different groups (e.g., age, gender, or cultural background) are affected remains unclear. We set out to approach these shortcomings in the music domain by posing the following research questions:

- *RQ1: Which novel insights into popularity bias can be obtained by quantifying algorithmic popularity bias based on the median, a variety of statistical moments, and similarity measures between popularity distributions?*
- *RQ2: Do algorithmic popularity biases affect users of different genders in the same way?*

We find that users of different genders are affected by algorithm-inflected bias differently, such that the majority of the models expose female users to more biased results. Also, algorithms less inclined towards popularity bias amplification do not necessarily perform worse in terms of utility (NDCG). Finally, the studied metrics provide novel insights into popularity bias in comparison with only using average differences.

2 RELATED WORK

We focus on popularity bias, a well-studied form of bias in recommender systems research. This form of bias refers to the underrepresentation of less popular items in the produced recommendations and can lead to a significantly worse recommendation quality for consumers of long tail or niche items [3, 10, 12, 13]. Abdollahpouri et al. [3] show that state-of-the-art movie recommendation algorithms suffer from popularity bias, and introduce the delta-GAP metric to quantify the level of underrepresentation. As shown in Kowald et al. [12], in particular users interested in niche, unpopular items suffer from a worse recommendation quality. The authors use the delta-GAP metric in the domain of music recommendations, and find that the delta-GAP metric does not show a difference between “niche” and “mainstream” users. The reason for this could be that a group-based metric is not suitable for the complexity of music styles, as user groups can be quite diverse within themselves [11]. Zhu et al. [20] address a related problem of item under-recommendation bias, expressing it with ranking-based statistical parity and ranking-based equal opportunity metrics. Boratto et al. [4] propose metrics quantifying the degree to which a recommender equally treats items along the popularity tail.

In contrast to these works, we study differences between popularity distributions of consumed and recommended items for each user. We express them in terms of the median as well as several statistical moments and similarity measures. In addition, we combine research strands on popularity bias and gender bias by analyzing how female and male listeners are affected by popularity bias.

3 MEASURING POPULARITY BIAS

We introduce ways to express popularity bias as quantified dissimilarity between popularity distributions of recommended and consumed items for each user.

3.1 Track Popularity Distributions

We define $P(t)$ popularity of a track t as the sum of its play counts over all users $u_i \in U$ in the dataset, namely $P(t) = \sum_{u_i \in U} PC(t, u_i)$. We then use these popularity estimates to derive the popularity distribution over each user’s listening history and recommendation list. In order to make the popularity distribution $H_{u_i}(t)$ over a user’s listening history $T_{hist}(u_i)$ comparable to the respective distribution $R_{u_i}(t)$ over the recommendation lists, we consider only the top of the recommendation list $T_{top_rec}(u_i)$ so that its length (number of tracks) matches the length of the user’s listening history $|T_{top_rec}(u_i)| = |T_{hist}(u_i)|$. Therefore, we define the popularity distribution over the listening history and the recommendation list of user u_i as follows:

$$H_{u_i}(t) = \begin{cases} P(t) & |t \in T_{hist}(u_i) \\ 0 & |t \notin T_{hist}(u_i) \end{cases} \quad R_{u_i}(t) = \begin{cases} P(t) & |t \in T_{top_rec}(u_i) \\ 0 & |t \notin T_{top_rec}(u_i) \end{cases} \quad (1)$$

To gain a better understanding of these distributions, Figure 1a shows an example of popularity distributions over a user’s listening history $T_{hist}(u_i)$ and the corresponding recommendation list $T_{top_rec}(u_i)$ produced by the SLIM recommender algorithm.

3.2 Metrics

3.2.1 Delta Metrics of Popularity Bias. In order to measure the differences between these distributions, we first introduce a series of delta metrics to calculate the discrepancies between the listening history and recommendation list popularity distributions of each user, and then aggregate them to achieve per-system results. We study five $\% \Delta M$ (percent delta) metrics where the metric M is one of the following: *Mean*, *Median*, *Variance*, *Skew*, *Kurtosis*. If $M(H_{u_i}(t))$ and $M(R_{u_i}(t))$ are the results of application of the same metric M to the two respective distributions, the respective $\% \Delta M$ for the user u_i is calculated as: $\% \Delta M_{u_i} = \frac{M(R_{u_i}(t)) - M(H_{u_i}(t))}{M(H_{u_i}(t))} \cdot 100$

Positive $\% \Delta Mean$ and $\% \Delta Median$ indicate that overall more popular tracks are recommended to the user. Since *Mean* is sensitive to outliers, the interplay between these metrics provides additional information about the changes in popularity. Positive $\% \Delta Variance$ means that the list of recommended items is more diverse in terms of different popularity values than the user’s history. This can also mean an increase in bias towards more popular items, as the most popular items are sparsely distributed across the popularity range. Positive $\% \Delta Skew$ denotes that the right tail of the recommendation list distribution is heavier (with respect to the left tails) than the one belonging to the user-history distribution. A positive value therefore means that more items tend to have lower popularity from the range of the distribution. Finally, positive $\% \Delta Kurtosis$ shows that the tails of the recommended tracks’ popularity distribution are heavier than of its counterpart, and the distribution itself is in a way closer to uniform distribution.

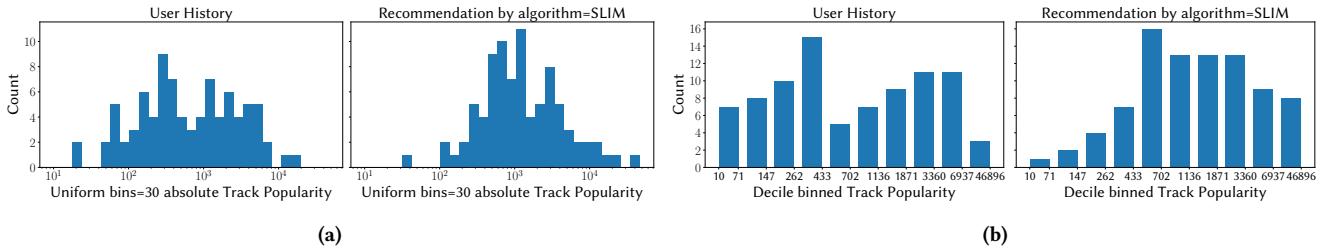


Figure 1: (a) shows equally binned (for visualization purposes only) distributions of popularity over the listening history (left) and the recommendation list (right) for the same user. On x-axis evenly binned popularity, on y-axis number of tracks in the distribution, falling into each bin. (b) demonstrates the same distributions binned with respect to the popularity distribution in the whole collection. This binning is employed for KL and Kendall's τ calculations.

Finally, the discussed metrics describe the difference between the distributions for a particular user. In order to represent the change across all users, we take the median of the per-user values.

3.2.2 Kullback–Leibler Divergence and Kendall's τ as Measures for Popularity Bias. In order to compare the entire popularity distributions, we utilize Kullback–Leibler Divergence (KL) and Kendall's τ (KT). For each user, we apply these metrics to the corresponding $H_{u_i}(t)$ and $R_{u_i}(t)$ decile-binned with respect to the popularity distribution over the whole collection ($P(t)$). The bins are chosen in such a way that the cumulative popularity of all tracks of the collection belonging into one bin constitutes approximately 10% of the total popularity of all tracks of the whole collection. Figure 1b shows the distributions from Figure 1a binned this way. In our dataset, the bin corresponding to the most popular tracks is constituted by only 161 items whose popularity ranges from about 7k to 47k total play counts. Each bin covers items that are roughly half as popular as the next decile bin and two times as popular as the previous decile bin. Such binning allows the two metrics to be less sensitive to minor differences between the distributions and concentrate on the shifts between different popularity categories.

KL estimates the dissimilarity of two distributions, in our case, between the user's listening history and recommendation list popularity distributions. It is defined as

$$KL_{u_i}(\hat{H}_{u_i}(b) | \hat{R}_{u_i}(b)) = \sum_{b_j \in B} \hat{H}_{u_i}(b_j) \log \frac{\hat{H}_{u_i}(b_j)}{\hat{R}_{u_i}(b_j)} \quad (2)$$

where $\hat{H}_{u_i}(b)$ and $\hat{R}_{u_i}(b)$ are decile-binned and normalized versions of the distributions and $b_j \in B$ represent the ten bins. KL compares the two distributions and increases with every mismatch in the item counts. It is particularly sensitive to the case when for a bin the user gets recommended fewer tracks than they have in their listening history.

While KL Divergence is sensitive to actual count changes, Kendall's τ metric reflects whether the order of bins is the same for the two distributions when ranked according to the respective counts. Kendall's τ is calculated as $KT_{u_i}(\hat{H}_{u_i}(b), \hat{R}_{u_i}(b)) = \frac{C-D}{C+D}$, where C represents the number of pairs of bins that have the same respective ranking in both distributions (concordant pairs) and D the number of pairs of bins that have the different respective ranking in the two distribution (discordant pairs). For example, looking at Figure 1b, the first two bins are concordant ($\in C$) as in both cases, more items fall into the second bin. While the first and the last bins

are discordant ($\in D$) as in the listening history distribution, the first bin has more items. However, the recommended distribution shows the opposite. This way, KT shows whether there are common patterns (correlations) in the two distributions, and it reaches its maximum value of 1 when the two distributions are identical from the bin-ranking point of view. Similar to $\%AM$ metrics, we use the median of the per-user values to measure the differences across all users for KL and KT .

4 EXPERIMENT SETUP

4.1 Recommendation Algorithms

To study algorithmic popularity biases, we examine different commonly used collaborative filtering algorithms (i.e., heuristic, neighborhood based, matrix factorization, and autoencoders) [6, 16]:

- Random Item (RAND): A baseline algorithm that recommends for each user random items. It avoids recommending already consumed items.
- Most Popular Items (POP): A baseline that implements a heuristic-based algorithm that recommends the same set of overall most popular items to each user.
- Item k-Nearest Neighbors (ItemKNN) [7]: A neighborhood-based algorithm that recommends items based on item-to-item similarity. Specifically, an item is recommended to a user if the item is similar to the items previously selected by the user. ItemKNN uses statistical measures to compute the item-to-item similarities.
- Sparse Linear Method (SLIM) [17]: Also a neighborhood-based algorithm, but instead of using predefined similarity metrics, the item-to-item similarity is learned directly from the data with a regression model.
- Alternating Least Squares (ALS) [9]: A matrix factorization approach that learns user and item embeddings such that the dot product of these two approximates the original user-item interaction matrix.
- Matrix factorization with Bayesian Personalized Ranking (BPR) [18]: Learns user and item embeddings, however, with an optimization function that aims to rank the items consumed by the users according to their preferences (hence, personalized ranking) instead of predicting the rating for a specific pair of user and item.
- Variational Autoencoder (VAE) [14]: An autoencoder-based algorithm that, given the user's interaction vector, estimates

Table 1: Statistics of the dataset. Number of Users, Tracks and listening events (LEs) are reported across F(emale) and M(ale) separately and also together (All). Mean and standard deviation (indicated after \pm) of the interactions of users with tracks and LEs are indicated in the last three columns, respectively.

Gender	Users	Tracks	LEs	Tracks/User	LEs/User
All	19,972	99,831	19,906,272	142 ± 172	$997 \pm 1,571$
F	4,415	70,980	3,397,310	101 ± 121	$769 \pm 1,158$
M	15,557	99,810	16,508,962	153 ± 182	$1,061 \pm 1,664$

a probability distribution over all the items using a variational autoencoder architecture.

For training the models, we use the same hyperparameter settings as provided by Melchiorre et al. [16].

4.2 Dataset and Evaluation Protocol

We perform experiments on *LFM-2b-DemoBias* [16], a subset of the *LFM-2b* dataset¹. As in [16], we only consider user-track interactions with a playcount (PC) > 1 , possibly avoiding using spurious interactions likely introduced by noise. Furthermore, we only consider tracks listened to by at least 5 different users and, likewise, only users who listened to at least 5 different tracks. Moreover, we only consider listening events within the last 5 years, letting us focus more on possible popularity biases in the recent years. Lastly, we consider binary user-track interactions, i.e., 1 if the user has listened to the track at least once, 0 otherwise.

The procedure described above results in a subset of 23k users over 1.6 million items. We finalize data preparation by sampling 100k tracks uniformly-at-random, which ensures that tracks of different popularity levels are equally likely to be included in the final dataset. The statistics of the final dataset are reported in Table 1. We find that males represent the majority group in the dataset and that they create $\sim 80\%$ of all listening events.

As evaluation protocol, we employ a user-based split strategy [14, 15], i.e., we split the 19,972 users in the dataset into train, validation, and test user groups via a 60-20-20 ratio split. We carry out 5-fold cross validation and change these user groups in a round-robin fashion. The users in the training set and all their interactions are used to train the recommendation algorithms. For testing and validation, we follow standard setups [14, 19] and randomly sample 80% of the users' items as input for the recommendation models and use the remaining 20% to calculate the evaluation metric.

5 RESULTS AND DISCUSSION

The results are shown in Table 2. Each value in the *All* rows, regarding the popularity bias metrics, shows the median value of the distribution of a given metric over all users. For instance, $\%Var$ of 72.6% for ALS denotes that the median increase in popularity variance is 72.6 percent between user's listening history and items recommended to each user across all users. SLIM KL 1.66 expresses that the median difference between user history popularity distributions and the corresponding recommended tracks popularity distributions is 1.66 in terms of KL Divergence. The reported results

regarding the genders indicate the changes in values in respect to the *All* values.

Both baseline algorithms (RAND and POP) show poor results on accuracy metrics. Notably, on the $\% \Delta$ popularity bias metrics, they show divergent behavior. Decreasing of $\% \Delta$ metrics of *Mean*, *Median*, *Variance* and increasing of *Skew* and *Kurtosis* indicate that RAND provides a list of tracks whose popularity distribution is closer to uniform than those from users' listening histories. POP has an opposite trend, as the recommended tracks' popularity distribution has a more pronounced peak, is skewed, and shifted towards more popular items. It also shows a substantial median increase of variance in popularity, which can be explained by the fact that in our dataset, the most popular tracks are sparsely distributed across a wide range of popularity values (161 track in the popularity range between 7k and 47k of total play counts). Thus, recommending tracks from this category leads to a high variance. High values for *KL* for both baselines also indicate that the overall popularity distributions of the recommended items are highly different from those of the users' listening histories. The random recommender demonstrates a higher median Kendall's τ , which means that its output better correlates with users' histories in terms of popularity distribution. Both neighborhood-based models (i.e., ItemKNN and SLIM) show a high performance in terms of NDCG and a moderate popularity bias in their recommendations according to the $\% \Delta$ metrics, which is lower compared to VAE and ALS. In particular, SLIM shows higher value in $\% \Delta$ *Mean* and *Median* compared to ItemKNN, suggesting that the item-to-item similarities learned by SLIM favors more popular items in the recommendations. ItemKNN displays lower *KL* and higher Kendall's τ than SLIM, which means that its results better approximate users' listening histories (we attribute this to ItemKNN being less sensitive to bias in the data as it does not require trainable parameters). These observations regarding the performance of the models indicate that a decrease in popularity bias does not necessarily lead to a significant performance drop. Comparing ALS with BPR, we can observe an opposite behavior. While providing less biased results, BPR shows the poorest performance among all non-baseline algorithms. While VAE is similarly biased in terms of all metrics as POP, it achieves a higher performance according to NDCG.

Comparing metrics between the two gender groups, we note that $\% \Delta$ *Mean* and *Median* is higher for female users. That means that their recommendations contain more popular items and/or items of higher popularity than the ones they usually listen to, and for this user group, that effect is more pronounced (hence larger values). Considering that $\% \Delta$ *Variance* is lower for the female users, we conclude that their recommendations are less diverse in

¹<http://www.cp.jku.at/datasets/LFM-2b>

Table 2: Results of algorithm-inflected popularity bias evaluation in terms of the seven introduced metrics and NDCG@10. Each model is represented by three rows. The row *All* gives the results on the whole dataset. The rows ΔFemale and ΔMale describe the difference in the result between the user group and the whole population in the dataset. For example, the $\% \Delta \text{Variance}$ for algorithm SLIM for *All* of 56.0 denotes a median increase in popularity variance (between listening history and recommended list) of 56% over all users. The corresponding ΔFemale value of -17.4 means that the variance increase for this group is $56.0 - 17.4 = 38.6\%$.

Alg.	Users	% ΔMean	% ΔMedian	% $\Delta\text{Var.}$	% ΔSkew	% $\Delta\text{Kurtosis}$	KL	Kendall's τ	NDCG@10
RAND	<i>All</i>	-91.8	-87.2	-99.5	11.5	15.3	3.904	0.165	0.000
	ΔFemale	-1.8	-3.5	-0.2	+0.0	-3.5	+0.976	-0.189	-0.000
	ΔMale	+0.5	+1.1	+0.1	-0.0	+1.3	-0.281	+0.053	+0.000
POP	<i>All</i>	432.5	975.2	487.0	-58.0	-87.0	6.023	0.057	0.045
	ΔFemale	+11.0	+282.1	-172.2	-2.1	-1.9	+1.626	-0.033	+0.003
	ΔMale	-2.8	-115.8	+55.9	+0.5	+0.5	-0.380	+0.016	-0.001
ALS	<i>All</i>	121.8	316.6	72.6	-25.2	-43.9	4.368	0.046	0.184
	ΔFemale	+9.9	+27.4	-7.1	-3.2	-5.4	+0.467	+0.110	-0.017
	ΔMale	-2.7	-6.6	+1.6	+0.8	+1.5	-0.121	-0.023	+0.005
BPR	<i>All</i>	-49.0	-3.7	-87.4	-14.8	-29.4	1.202	0.268	0.129
	ΔFemale	+5.2	+7.7	+2.1	-1.4	-3.9	+0.476	-0.043	-0.011
	ΔMale	-1.1	-1.9	-0.6	+0.4	+1.1	-0.110	+0.010	+0.003
ItemKNN	<i>All</i>	9.6	4.6	5.7	-14.3	-29.0	0.175	0.423	0.301
	ΔFemale	+2.0	+5.8	-2.6	-2.1	-3.2	+0.128	-0.037	-0.042
	ΔMale	-0.5	-1.3	+0.9	+0.8	+0.9	-0.020	+0.008	+0.012
SLIM	<i>All</i>	49.8	99.8	56.0	-12.5	-26.0	0.424	0.189	0.365
	ΔFemale	-6.4	-13.1	-17.4	-1.7	-4.6	+0.217	+0.052	-0.048
	ΔMale	+1.9	+3.9	+5.6	+0.6	+1.1	-0.029	-0.012	+0.014
VAE	<i>All</i>	303.9	736.3	351.0	-45.2	-70.1	4.823	-0.028	0.191
	ΔFemale	+10.1	+56.4	-69.3	-6.2	-6.6	+0.633	+0.146	-0.020
	ΔMale	-2.3	-20.4	+17.3	+1.8	+2.1	-0.161	-0.042	+0.006

terms of track popularity while consisting of more popular items. Judging by $\% \Delta \text{Skew}$, Kurtosis as well as Kendall's τ , we can suggest that most recommender algorithms provide recommendations with comparable popularity distributions to both male and female users. At the same time, a slightly larger KL may mean a larger shift towards popular items for female users. ItemKNN is the least biased algorithm in our study. It features low absolute values of $\% \Delta \text{Mean}$, Median and Variance , meaning that its recommendations consist of tracks comparable to the user's listening history in terms of average popularity and variety. High Kendall's τ means that the shape of the popularity distribution of the recommendations best matches the user's history among all tested algorithms. Still, it is slightly biased towards more popular items, as shown by negative $\% \Delta \text{Skew}$ and KL (which combined with high Kendall's τ signalizes about a shift of the distribution).

6 CONCLUSIONS AND FUTURE DIRECTION

In this paper, we examine to what extent various music recommender systems amplify item popularity bias. We study seven metrics of popularity bias deviation and analyze the results of seven recommender algorithms for users of different genders and for the overall population in the dataset. Addressing *RQ1*, we observe that the studied metrics capture considerably different aspects of difference between popularity distributions of consumed and recommended items. While $\% \Delta \text{Mean}$ and $\% \Delta \text{Median}$ tell us about

overall trends (are recommended tracks more or less popular than consumed ones), $\% \Delta \text{Variance}$ expresses the change in the diversity between listening histories and recommendation lists, and $\% \Delta \text{Skew}$ and $\% \Delta \text{Kurtosis}$ hint on the difference of shapes between the two distributions. Finally, KL Divergence and Kendall's τ allow insight into how well the distributions match on a more granular level. With regard to *RQ2*, we found that while the investigated algorithms display various levels of popularity bias, the majority of them (VAE, ItemKNN, BPR, ALS) expose the female users to more popularity biased results. In the future, we will approach mitigating model-imposed popularity bias, e.g., through adversarial training or incorporating bias into the loss function of the recommenders, as well as finding more expressive metrics describing differences in the popularity distributions. Additionally, we plan to split our users into groups according to mainstreaminess as in [12] to compare our metrics with the group-based delta-GAP metric used in that work.

ACKNOWLEDGMENTS

This work was funded by the H2020 project AI4EU (GA: 825619), the Austrian Science Fund (FWF): P33526, and the FFG COMET program.

REFERENCES

- [1] Himan Abdollahpour, Robin Burke, and Bamshad Mobasher. 2017. Controlling popularity bias in learning-to-rank recommendation. In *Proceedings of the eleventh*

- ACM conference on recommender systems.* 42–46.
- [2] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2019. Managing popularity bias in recommender systems with personalized re-ranking. In *The thirty-second international flairs conference*.
 - [3] Himan Abdollahpouri, Masoud Mansouri, Robin Burke, and Bamshad Mobasher. 2019. The Unfairness of Popularity Bias in Recommendation. In *Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019), Copenhagen, Denmark, September 20, 2019 (CEUR Workshop Proceedings, Vol. 2440)*. CEUR-WS.org. <http://ceur-ws.org/Vol-2440/paper4.pdf>
 - [4] Ludovico Boratto, Gianni Fenu, and Mirko Marras. 2021. Connecting user and item perspectives in popularity debiasing for collaborative recommendation. *Information Processing & Management* 58, 1 (Jan 2021), 102387. <https://doi.org/10.1016/j.ipm.2020.102387>
 - [5] Óscar Celma. 2010. *Music Recommendation and Discovery - The Long Tail, Long Fail, and Long Play in the Digital Music Space*. Springer. <https://doi.org/10.1007/978-3-642-13287-2>
 - [6] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 101–109.
 - [7] Mukund Deshpande and George Karypis. 2004. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)* 22, 1 (2004), 143–177.
 - [8] Michael D Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. 2018. All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In *Conference on Fairness, Accountability and Transparency*. 172–186.
 - [9] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*. Ieee, 263–272.
 - [10] Dietmar Jannach, Lukas Lerche, and Iman Kamehkhosh. 2015. Beyond "Hitting the Hits": Generating Coherent Music Playlist Continuations with the Right Tracks. In *Proceedings of the 9th ACM Conference on Recommender Systems* (Vienna, Austria) (RecSys '15). ACM, New York, NY, USA, 187–194. <https://doi.org/10.1145/2792838.2800182>
 - [11] Dominik Kowald, Peter Muellner, Eva Zangerle, Christine Bauer, Markus Schedl, and Elisabeth Lex. 2021. Support the underground: characteristics of beyond-mainstream music listeners. *EPJ Data Science* 10, 1 (2021), 1–26.
 - [12] Dominik Kowald, Markus Schedl, and Elisabeth Lex. 2020. The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12036)*. Springer, 35–42. https://doi.org/10.1007/978-3-030-45442-5_5
 - [13] Elisabeth Lex, Dominik Kowald, and Markus Schedl. 2020. Modeling popularity and temporal drift of music genre preferences. *Transactions of the International Society for Music Information Retrieval* 3, 1 (2020).
 - [14] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*. 689–698.
 - [15] Benjamin Marlin. 2004. *Collaborative filtering: A machine learning perspective*. University of Toronto Toronto.
 - [16] Alessandro B. Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. 2021. Investigating gender fairness of recommendation algorithms in the music domain. *Information Processing & Management* 58, 5 (2021), 102666. <https://doi.org/10.1016/j.ipm.2021.102666>
 - [17] Xia Ning and George Karypis. 2011. Slim: Sparse linear methods for top-n recommender systems. In *2011 IEEE 11th International Conference on Data Mining*. IEEE, 497–506.
 - [18] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).
 - [19] Harald Steck. 2019. Embarrassingly Shallow Autoencoders for Sparse Data. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019*. Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia (Eds.). ACM, 3251–3257. <https://doi.org/10.1145/3308558.3313710>
 - [20] Ziwei Zhu, Jianling Wang, and James Caverlee. 2020. *Measuring and Mitigating Item Under-Recommendation Bias in Personalized Ranking Systems*. Association for Computing Machinery, New York, NY, USA, 449–458. <https://doi.org/10.1145/3397271.3401177>

P14 Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems (2022)

Fairness and Popularity Bias in Recommender Systems

P14 Kowald, D., Lacic, E. (2022). Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2022)*. Communications in Computer and Information Science, vol. 1610, pp. 1-11.

DOI: https://doi.org/10.1007/978-3-031-09316-6_1



Popularity Bias in Collaborative Filtering-Based Multimedia Recommender Systems

Dominik Kowald^{1,2} (✉) and Emanuel Lacic¹

¹ Know-Center GmbH, Graz, Austria

{dkowald,elacic}@know-center.at

² Graz University of Technology, Graz, Austria

Abstract. Multimedia recommender systems suggest media items, e.g., songs, (digital) books and movies, to users by utilizing concepts of traditional recommender systems such as collaborative filtering. In this paper, we investigate a potential issue of such collaborative-filtering based multimedia recommender systems, namely popularity bias that leads to the underrepresentation of unpopular items in the recommendation lists. Therefore, we study four multimedia datasets, i.e., Last.fm, MovieLens, BookCrossing and MyAnimeList, that we each split into three user groups differing in their inclination to popularity, i.e., LowPop, MedPop and HighPop. Using these user groups, we evaluate four collaborative filtering-based algorithms with respect to popularity bias on the item and the user level. Our findings are three-fold: firstly, we show that users with little interest into popular items tend to have large user profiles and thus, are important data sources for multimedia recommender systems. Secondly, we find that popular items are recommended more frequently than unpopular ones. Thirdly, we find that users with little interest into popular items receive significantly worse recommendations than users with medium or high interest into popularity.

Keywords: multimedia recommender systems · collaborative filtering · popularity bias · algorithmic fairness

1 Introduction

Collaborative filtering (CF) is one of the most traditional but also most powerful concepts for calculating personalized recommendations [22] and is vastly used in the field of multimedia recommender systems (MMRS) [11]. However, one issue of CF-based approaches is that they are prone to popularity bias, which leads to the overrepresentation of popular items in the recommendation lists [2,3]. Recent research has studied popularity bias in domains such as music [15,16] or movies [3] by comparing the recommendation performance for different user groups that differ in their inclination to mainstream multimedia items. However, a comprehensive study of investigating popularity bias on the item and user level across several multimedia domains is still missing (see Sect. 2).

In the present paper, we therefore build upon these previous works and expand the study of popularity bias to four different domains of MMRS: music (Last.fm), movies (MovieLens), digital books (BookCrossing), and animes (MyAnimeList). Within these domains, we show that users with little interest into popular items tend to have large user profiles and thus, are important consumers and data sources for MMRS. Furthermore, we apply four different CF-based recommendation algorithms (see Sect. 3) on our four datasets that we each split into three user groups that differ in their inclination to popularity (i.e., LowPop, MedPop, and HighPop). With this, we address two research questions (RQ):

- **RQ1:** To what extent does an item’s popularity affect this item’s recommendation frequency in MMRS?
- **RQ2:** To what extent does a user’s inclination to popular items affect the quality of MMRS?

Regarding **RQ1**, we find that the probability of a multimedia item to be recommended strongly correlates with this items’ popularity. Regarding **RQ2**, we find that users with less inclination to popularity (LowPop) receive statistically significantly worse multimedia recommendations than users with medium (MedPop) and high (HighPop) inclination to popular items (see Sect. 4). Our results demonstrate that although users with little interest into popular items tend to have the largest user profiles, they receive the lowest recommendation accuracy. Hence, future research is needed to mitigate popularity bias in MMRS, both on the item and the user level.

2 Related Work

This section presents research on popularity bias that is related to our work. We split these research outcomes in two groups: (i) work related to recommender systems in general, and (ii) work that focuses on popularity bias mitigation techniques.

Popularity Bias in Recommender Systems. Within the domain of recommender systems, there is an increasing number of works that study the effect of popularity bias. For example, as reported in [8], bias towards popular items can affect the consumption of items that are not popular. This in turn prevents them to become popular in the future at all. That way, a recommender system is prone to ignoring novel items or the items liked by niche users that are typically hidden in the “long-tail” of the available item catalog. Tackling these long-tail items has been recognized by some earlier work, such as [10, 20]. This issue is further investigated by [1, 2] using the popular movie dataset MovieLens 1M. The authors show that more than 80% of all ratings actually belong to popular items, and based on this, focus on improving the trade-off between the ranking accuracy and coverage of long-tail items. Research conducted in [13] illustrates a comprehensive algorithmic comparison with respect to popularity bias. The authors analyze multimedia datasets such as MovieLens, Netflix, Yahoo!Movies and BookCrossing, and find that recommendation methods only consider a small fraction of

the available item spectrum. For instance, they find that KNN-based techniques focus mostly on high-rated items and factorization models lean towards recommending popular items. In our work, we analyze an even larger set of multimedia domains and study popularity bias not only on the item but also on the user level.

Popularity Bias Mitigation Techniques. Typical research on mitigating popularity bias performs a re-ranking step on a larger set of recommended candidate items. The goal of such post-processing approaches is to better expose long-tail items in the recommendation list [2, 4, 6]. Here, for example, [7] proposes to improve the total number of distinct recommended items by defining a target distribution of item exposure and minimizing the discrepancy between exposure and recommendation frequency of each item. In order to find a fair ratio between popular and less popular items, [24] proposes to create a protected group of long-tail items and to ensure that their exposure remains statistically indistinguishable from a given minimum. Beside focusing on post-processing, there are some in-processing attempts in adapting existing recommendation algorithms in a way that the generated recommendations are less biased toward popular items. For example, [5] proposes to use a probabilistic neighborhood selection for KNN methods, or [23] suggests a blind-spot-aware matrix factorization approach that debiases interactions between the recommender system and the user. We believe that the findings of our paper can inform future research on choosing the right mitigation technique for a given setting.

3 Method

In this section, we describe (i) our definition of popularity, (ii) our four multimedia datasets, and (iii) our four recommendation algorithms based on collaborative filtering as well as our evaluation protocol.

3.1 Defining Popularity

Here, we describe how we define popularity (i) on the item level, and (ii) on the user level. We use the item popularity definition of [3], where the item popularity score Pop_i of an item i is given by the relative number of users who have rated i , i.e., $Pop_i = \frac{|U_i|}{|U|}$. Based on this, we can also define $Pop_{i,u}$ as the average item popularity in the user profile I_u , i.e., $Pop_{i,u} = \frac{1}{|I_u|} \sum_{i \in I_u} Pop_i$. Additionally, we can also define an item i as popular if it falls within the top-20% of item popularity scores. Thus, we define $I_{u,Pop}$ as the set of popular items in the user profile.

On the user level, we also follow the work of [3] and define a user u 's inclination to popularity Pop_u as the ratio of popular items in the user profile, i.e., $Pop_u = \frac{|I_{u,Pop}|}{|I_u|}$. As an example, $Pop_u = 0.8$ if 80% of the items in the user's item history are popular ones. We use this definition to create the LowPop, MedPop and HighPop user groups in case of MovieLens, BookCrossing and MyAnimeList.

Table 1. Statistics of our four datasets, where $|U|$ is the number of users, $|I|$ is the number of media items, $|R|$ is the number of ratings, sparsity is defined as the ratio of observed ratings $|R|$ to possible ratings $|U| \times |I|$, and R -range is the rating range.

Dataset	$ U $	$ I $	$ R $	$ R / U $	$ R / I $	Sparsity	R -range
Last.fm	3,000	352,805	1,755,361	585	5	0.998	[1–1,000]
MovieLens	3,000	3,667	675,610	225	184	0.938	[1–5]
BookCrossing	3,000	223,607	577,414	192	3	0.999	[1–10]
MyAnimeList	3,000	9,450	649,814	216	69	0.977	[1–10]

In case of Last.fm, we use a definition for Pop_u especially proposed for the music domain, which is termed the mainstreaminess score [9]. Here, we use the $M_{R,APC}^{global}$ definition, which is already provided in the dataset¹ published in our previous work [16]. Formally, $M_{R,APC}^{global}(u) = \tau(ranks(APC), ranks(APC(u)))$, where APC and $APC(u)$ are the artist play counts averaged over all users and for a given user u , respectively. τ indicates the rank-order correlation according to Kendall's τ . Thus, u 's mainstreaminess score is defined as the overlap between a user's item history and the aggregated item history of all Last.fm users in the dataset. Thus, the higher the mainstreaminess score, the higher a user's inclination to popular music. Please note that we cannot calculate the mainstreaminess score for the other datasets, since we do not have multiple interactions per item (i.e., play counts) in these cases (only one rating per user-item pair).

To get a better feeling of the relationship between average item popularity scores in the user profiles (i.e., $Pop_{u,i}$) and the user profile size (i.e., $|I_u|$), we plot these correlations for our four datasets and per user group in Fig. 1. Across all datasets, we see a negative correlation between average item popularity and user profile size, which means that users with little interest in popular items tend to have large user profiles. This suggests that these users are important consumers and data sources in MMRS, and thus, should also be treated in a fair way (i.e., should receive similar accuracy scores as users with medium or high interest in popular items).

3.2 Multimedia Datasets

For our study, we use four datasets containing rating data of users for media items. The statistics of our datasets can be found in Table 1, and we provide the datasets via Zenodo². The users in each of our four datasets are split into three equally-sized user groups: (i) LowPop, i.e., the 1,000 users with the least inclination to popular items, (ii) MedPop, i.e., 1,000 users with medium inclination to popular media items, and (iii) HighPop, i.e., the 1,000 users with the highest inclination to popular media items. This sums up to $|U| = 3,000$ users per

¹ <https://zenodo.org/record/3475975>.

² <https://zenodo.org/record/6123879>.

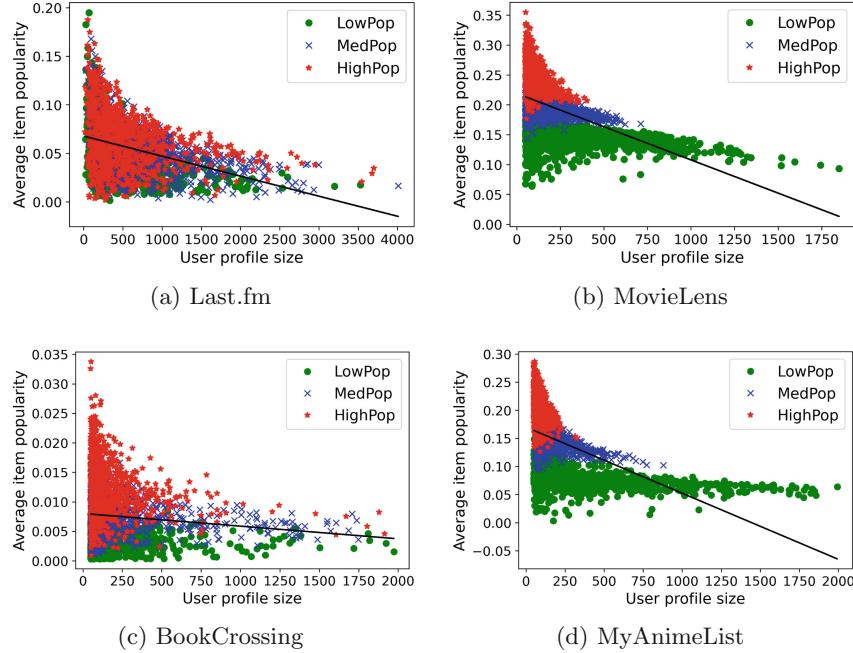


Fig. 1. Relationship between average item popularity scores in the user profiles (i.e., $Pop_{u,i}$) and user profile size (i.e., $|I_u|$). We see that users with little interest in popular items tend to have large user profiles.

dataset. Next, we describe our four datasets and how we split the user groups based on the popularity definitions given before:

Last.fm. For the music streaming platform Last.fm, we use the dataset published in our previous work [16], which is based on the LFM-1b dataset³. Here, a user is assigned to one of the three groups LowPop, MedPop and HighPop based on the user's mainstreamness score [9], which we defined earlier (i.e., $M_{R,APC}^{global}$). Additionally, in this Last.fm dataset, the listening counts of users for music artists are scaled to a rating range of [1–1,000]. When looking at Table 1, Last.fm has the largest number of items $|I| = 352,805$ and the largest number of ratings $|R| = 1,755,361$ across our four datasets.

MovieLens. In case of the movie rating portal MovieLens, we use the well-known MovieLens-1M dataset⁴. We extract all users with a minimum of 50 ratings and a maximum of 2,000 ratings. We assign these users to one of the three user groups LowPop, MedPop and HighPop based on the ratio of popular items in the user profiles [3] as described earlier (i.e., Pop_u). Table 1 shows that MovieLens is the least sparse (i.e., most dense) dataset in our study and also has the highest number of ratings per items ($|R|/|I|$).

³ <http://www.cp.jku.at/datasets/LFM-1b/>.

⁴ <https://grouplens.org/datasets/movielens/1m/>.

BookCrossing. The dataset of the (digital) book sharing platform BookCrossing was provided by Uni Freiburg⁵. We use the same popularity definitions, group assignment method as well as rating thresholds as in case of MovieLens. However, in contrast to MovieLens, BookCrossing contains not only explicit feedback in the form of ratings but also implicit feedback when a user bookmarks a book. In this case, we set the implicit feedback to a rating of 5, which is the middle value in BookCrossing’s rating range of [1–10]. Across all datasets, BookCrossing is the dataset with the highest sparsity.

MyAnimeList. We apply the same processing methods as used in case of BookCrossing to the MyAnimeList dataset, which is provided via Kaggle⁶. Similar to BookCrossing, MyAnimeList also contains implicit feedback when a user bookmarks an Anime, and again we convert this feedback to an explicit rating of 5, which is the middle value in the rating range.

3.3 Recommendation Algorithms and Evaluation Protocol

We use the same set of personalized recommendation algorithms as used in our previous work [16] but since we focus on CF-based methods, we replace the UserItemAvg algorithm with a scalable co-clustering-based approach [12] provided by the Python-based Surprise framework⁷. Thus, we evaluate two KNN-based algorithms without and with incorporating the average rating of the target user and item (UserKNN and UserKNNAvg), one non-negative matrix factorization variant [19] (NMF) as well as the aforementioned CoClustering algorithm. In most cases, we stick to the default parameter settings as suggested by the Surprise framework and provide the detailed settings in our GitHub repository⁸.

We also follow the same evaluation protocol as used in our previous work [16] and formulate the recommendation task as a rating prediction problem, which we measure using the mean absolute error (MAE). However, instead of using only one 80/20 train-set split, we use a more sophisticated 5-fold cross-validation evaluation protocol. To test for statistical significance, we perform pairwise t-tests between LowPop and MedPop as well as between LowPop and HighPop since we are interested if LowPop is treated in an unfair way by the MMRS. We report statistical significance for LowPop only in cases in which there is a significant difference between LowPop and MedPop as well as between LowPop and HighPop for all five folds.

4 Results

We structure our results based on our two research questions. Thus, we first investigate popularity bias on the item level by investigating the relationship

⁵ <http://www2.informatik.uni-freiburg.de/~cziegler/BX/>.

⁶ <https://www.kaggle.com/CooperUnion/anime-recommendations-database>.

⁷ <http://surpriselib.com/>.

⁸ <https://github.com/domkowald/FairRecSys>.

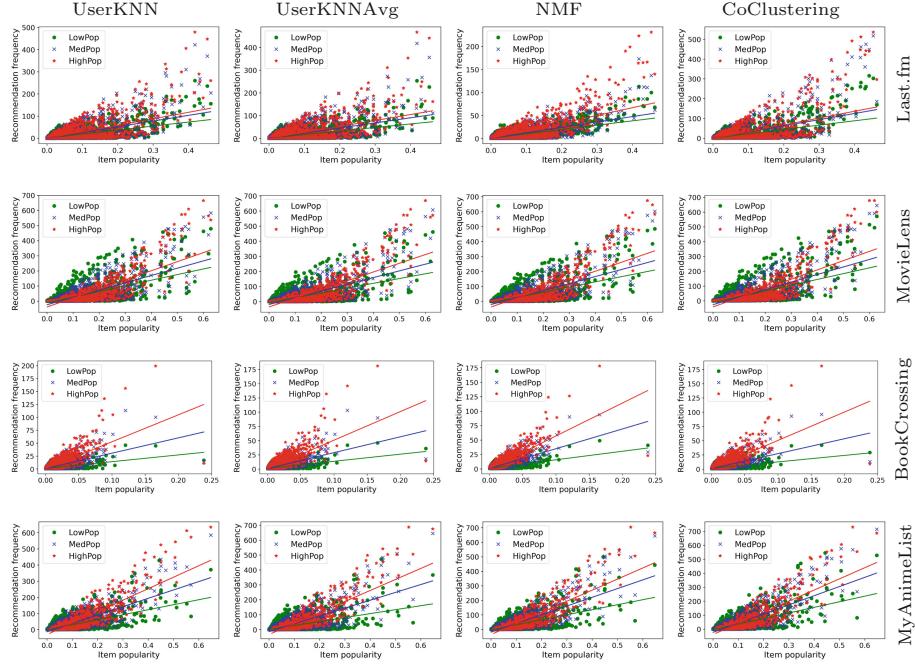


Fig. 2. RQ1: Relationship between item popularity and recommendation frequency of four CF-based algorithms for Last.fm, MovieLens, BookCrossing and MyAnimeList. In all 16 cases, we see that popular media items have a higher probability of being recommended than unpopular ones.

between item popularity and recommendation frequency (**RQ1**). Next, we investigate popularity bias on the user level by comparing the recommendation performance for our three user groups (**RQ2**).

4.1 RQ1: Relationship Between Item Popularity and Recommendation Frequency

Figure 2 shows the relationship between item popularity and recommendation frequency for the four CF-based algorithms UserKNN, UserKNNAvg, NMF and CoClustering on all five folds of our four multimedia datasets Last.fm, MovieLens, BookCrossing and MyAnimeList. The solid lines indicate the linear regression between the two variables for the three user groups.

In all 16 plots, and all three user groups, we observe a positive relationship between an item's popularity and how often this item gets recommended (**RQ1**). However, for NMF applied to Last.fm, the maximum recommendation frequency is much lower as in case of the other algorithms. Thus, only in case of NMF applied to Last.fm, we see a weak relationship between popularity and recommendation frequency, while in all other cases, we see a strong relationship

Table 2. RQ2: Mean absolute error (MAE) results (the lower, the better) of our study. The lowest accuracy is always given for the LowPop user group (statistically significant according to a t-test with $p < 0.001$ as indicated by $***$ and $p < 0.05$ as indicated by $**$). Across the algorithms, the best results are indicated by **bold numbers** and across the user groups, the best results are indicated by *italic numbers*.

Dataset	User group	UserKNN	UserKNNAvg	NMF	CoClustering
Last.fm	LowPop	49.489***	46.483***	39.641**	47.304***
	MedPop	<i>42.899</i>	<i>37.940</i>	32.405	37.918
	HighPop	45.805	43.070	38.580	42.982
MovieLens	LowPop	0.801***	0.763***	0.753***	0.738***
	MedPop	0.748	0.727	0.722	0.705
	HighPop	<i>0.716</i>	<i>0.697</i>	<i>0.701</i>	0.683
BookCrossing	LowPop	1.403***	1.372***	1.424***	1.392***
	MedPop	<i>1.154</i>	1.122	<i>1.214</i>	<i>1.134</i>
	HighPop	1.206	1.155	1.274	1.162
MyAnimeList	LowPop	1.373***	1.001***	1.010***	1.001***
	MedPop	1.341	0.952	0.968	<i>0.956</i>
	HighPop	<i>1.311</i>	0.948	<i>0.951</i>	0.975

between these variables. This is in line with our previous related work investigating popularity bias in Last.fm [16]. When comparing the three user groups, we see the weakest relationship between the variables for LowPop and the strongest relationship for HighPop. We will refer to this finding when investigating **RQ2**.

4.2 RQ2: Relationship Between Users’ Inclination to Popular Items and Recommendation Accuracy

Table 2 shows the MAE estimates for the aforementioned CF-based recommendation algorithms (UserKNN, UserKNNAvg, NMF, and CoClustering) on the four multimedia datasets (Last.fm, MovieLens, BookCrossing, and MyAnimeList) split in three user groups that differ in their inclination to popularity (LowPop, MedPop, and HighPop). Additionally, we indicate statistically significant differences between both LowPop and MedPop, and LowPop and HighPop according to a t-test with $p < 0.001$ using $***$ and with $p < 0.05$ using $**$ in the LowPop lines.

Across all datasets, we observe the highest MAE estimates, and thus lowest recommendation accuracy, for the LowPop user groups. The best results, indicated by *italic numbers*, are reached for the MedPop group in case of Last.fm and BookCrossing, and for the HighPop group in case of MovieLens and MyAnimeList. For Last.fm this is in line with our previous work [16]. Across the algorithms, we see varying results: for Last.fm, and again in line with our previous work [16], the best results are reached for NMF. For MovieLens, we get the best results for

the CoClustering approach, and for BookCrossing and MyAnimeList the highest accuracy is reached for the UserKNN variant UserKNNAvg. We plan to investigate these differences across the user groups and the algorithms in our future research, as outlined in the next section.

Taken together, users with little inclination to popular multimedia items receive statistically significantly worse recommendations by CF-based algorithms than users with medium and high inclination to popularity (**RQ2**). When referring back to our results of **RQ1** in Fig. 2, this is interesting since LowPop is the group with the weakest relationship between item popularity and recommendation frequency. However, this suggests that recommendations are still too popular for this user group and an adequate mitigation strategy is needed.

5 Conclusion

In this paper, we have studied popularity bias in CF-based MMRS. Therefore, we investigated four recommendation algorithms (UserKNN, UserKNNAvg, NMF, and CoClustering) for three user groups (LowPop, MedPop, and HighPop) on four multimedia datasets (Last.fm, MovieLens, BookCrossing, and MyAnimeList). Specifically, we investigated popularity bias from the item (**RQ1**) and user (**RQ2**) perspective. Additionally, we have shown that users with little interest into popular items tend to have large profile sizes, and therefore are important data sources for MMRS.

With respect to **RQ1**, we find that the popularity of a multimedia item strongly correlates with the probability that this item is recommended by CF-based approaches. With respect to **RQ2**, we find that users with little interest in popular multimedia items (i.e., LowPop) receive significantly worse recommendations than users with medium (i.e., MedPop) or high (i.e., HighPop) interest in popular items. This is especially problematic since users with little interest into popularity tend to have large profile sizes, and thus, should be treated in a fair way by MMRS.

Future Work. Our results demonstrate that future work should further focus on studying this underserved user group in order to mitigate popularity bias in CF-based recommendation algorithms. We believe that our findings are a first step to inform the research on popularity bias mitigation techniques (see Sect. 2) to choose the right mitigation strategy for a given setting.

Additionally, as mentioned earlier, we plan to further study the differences we found with respect to algorithmic performance for the different user groups and multimedia domains. Here, we also want to study popularity bias in top- n settings using ranking-aware metrics such as nDCG (e.g., as used in [18]). Finally, we plan to work on further bias mitigation strategies based on cognitive-inspired user modeling and recommendation techniques (e.g., [14, 17, 21]).

Acknowledgements. This research was funded by the H2020 project TRUSTS (GA: 871481) and the “DDAI” COMET Module within the COMET - Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for

Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG.

References

1. Abdollahpouri, H., Burke, R., Mobasher, B.: Controlling popularity bias in learning-to-rank recommendation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 42–46 (2017)
2. Abdollahpouri, H., Burke, R., Mobasher, B.: Managing popularity bias in recommender systems with personalized re-ranking. In: The Thirty-second International Flairs Conference (2019)
3. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation. In: RecSys Workshop on Recommendation in Multistakeholder Environments (RMSE) (2019)
4. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B., Malthouse, E.: User-centered evaluation of popularity bias in recommender systems. In: Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, pp. 119–129 (2021)
5. Adamopoulos, P., Tuzhilin, A.: On over-specialization and concentration bias of recommendations: probabilistic neighborhood selection in collaborative filtering systems. In: Proceedings of the 8th ACM Conference on Recommender Systems, pp. 153–160 (2014)
6. Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Trans. Knowl. Data Eng.* **24**(5), 896–911 (2011)
7. Antikacioglu, A., Ravi, R.: Post processing recommender systems for diversity. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 707–716 (2017)
8. Baeza-Yates, R.: Bias in search and recommender systems. In: Fourteenth ACM Conference on Recommender Systems, p. 2 (2020)
9. Bauer, C., Schedl, M.: Global and country-specific mainstreaminess measures: definitions, analysis, and usage for improving personalized music recommendation systems. *PLoS One* **14**(6), e0217389 (2019)
10. Brynjolfsson, E., Hu, Y.J., Smith, M.D.: From niches to riches: anatomy of the long tail. *Sloan Manage. Rev.* **47**(4), 67–71 (2006)
11. Deldjoo, Y., Schedl, M., Cremonesi, P., Pasi, G.: Recommender systems leveraging multimedia content. *ACM Comput. Surv. (CSUR)* **53**(5), 1–38 (2020)
12. George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: Fifth IEEE International Conference on Data Mining (ICDM 2005), p. 4. IEEE (2005)
13. Jannach, D., Lerche, L., Kamehkhosh, I., Jugovac, M.: What recommenders recommend: an analysis of recommendation biases and possible countermeasures. *User Model. User Adapt. Interact.* **25**(5), 427–491 (2015). <https://doi.org/10.1007/s11257-015-9165-3>
14. Kowald, D., Lex, E.: The influence of frequency, recency and semantic context on the reuse of tags in social tagging systems. In: Proceedings of Hypertext 2016, pp. 237–242. ACM, New York, NY, USA (2016)

15. Kowald, D., Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E.: Support the underground: characteristics of beyond-mainstream music listeners. *EPJ Data Sci.* **10**(1), 1–26 (2021). <https://doi.org/10.1140/epjds/s13688-021-00268-9>
16. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: a reproducibility study. In: Jose, J.M., et al. (eds.) ECIR 2020. LNCS, vol. 12036, pp. 35–42. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45442-5_5
17. Lacic, E., Kowald, D., Seitlinger, P.C., Trattner, C., Parra, D.: Recommending items in social tagging systems using tag and time information. In: Proceedings of the 1st Social Personalization Workshop co-located with the 25th ACM Conference on Hypertext and Social Media, pp. 4–9. ACM (2014)
18. Lacic, E., Kowald, D., Traub, M., Luzhnica, G., Simon, J.P., Lex, E.: Tackling cold-start users in recommender systems with indoor positioning systems. In: Poster Proceedings of the 9th ACM Conference on Recommender Systems. Association of Computing Machinery (2015)
19. Luo, X., Zhou, M., Xia, Y., Zhu, Q.: An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Trans. Indust. Inform.* **10**(2), 1273–1284 (2014)
20. Park, Y.J., Tuzhilin, A.: The long tail of recommender systems and how to leverage it. In: Proceedings of the 2008 ACM Conference on Recommender Systems, pp. 11–18 (2008)
21. Seitlinger, P., Kowald, D., Kopeinik, S., Hasani-Mavriqi, I., Lex, E., Ley, T.: Attention please! a hybrid resource recommender mimicking attention-interpretation dynamics. In: Proceedings of WWW 2015 companion, pp. 339–345. ACM (2015)
22. Shi, Y., Larson, M., Hanjalic, A.: Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges. *ACM Comput. Surv.* **47**(1), 3:1–3:45 (2014)
23. Sun, W., Khenissi, S., Nasraoui, O., Shafto, P.: Debiasing the human-recommender system feedback loop in collaborative filtering. In: Companion Proceedings of The 2019 World Wide Web Conference, pp. 645–651 (2019)
24. Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., Baeza-Yates, R.: Fa*ir: A fair top-k ranking algorithm. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1569–1578 (2017)

P15 What Drives Readership? Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations (2022)

Fairness and Popularity Bias in Recommender Systems

[P15] Lacic, E., Fadljevic, L., Weissenboeck, F., Lindstaedt, S., **Kowald, D.** (2022). What Drives Readership? An Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations. In *Proceedings of the 44th European Conference on Information Retrieval (ECIR'2022)*, pp. 172-179.

DOI: https://doi.org/10.1007/978-3-030-99739-7_20



What Drives Readership? An Online Study on User Interface Types and Popularity Bias Mitigation in News Article Recommendations

Emanuel Lacic¹, Leon Fadljevic¹, Franz Weissenboeck², Stefanie Lindstaedt¹, and Dominik Kowald^{1()}

¹ Know-Center GmbH, Graz, Austria

{elacic,lfadljevic,slind,dkowald}@know-center.at

² Die Presse Verlags-GmbH, Vienna, Austria

franz.weissenboeck@diepresse.com

Abstract. Personalized news recommender systems support readers in finding the right and relevant articles in online news platforms. In this paper, we discuss the introduction of personalized, content-based news recommendations on DiePresse, a popular Austrian online news platform, focusing on two specific aspects: (i) user interface type, and (ii) popularity bias mitigation. Therefore, we conducted a two-weeks online study that started in October 2020, in which we analyzed the impact of recommendations on two user groups, i.e., anonymous and subscribed users, and three user interface types, i.e., on a desktop, mobile and tablet device. With respect to user interface types, we find that the probability of a recommendation to be seen is the highest for desktop devices, while the probability of interacting with recommendations is the highest for mobile devices. With respect to popularity bias mitigation, we find that personalized, content-based news recommendations can lead to a more balanced distribution of news articles' readership popularity in the case of anonymous users. Apart from that, we find that significant events (e.g., the COVID-19 lockdown announcement in Austria and the Vienna terror attack) influence the general consumption behavior of popular articles for both, anonymous and subscribed users.

Keywords: News recommendation · User interface · Popularity bias

1 Introduction

Similar to domains such as social networks or social tagging systems [14, 17, 21], the personalization of online content has become one of the key drivers for news portals to increase user engagement and convince readers to become paying subscribers [8, 9, 22]. A natural way for news portals to do this, is to provide their users with articles that are fresh and popular. This is typically achieved via simple most-popular news recommendations, especially since this approach has been

shown to provide accurate recommendations in offline evaluation settings [11]. However, such an approach could amplify popularity bias with respect to users' news consumption. This means that the equal representation of non-popular, but informative content in the recommendation lists is put into question, since articles from the "long tail" do not have the same chance of being represented and served to the user [1]. Since nowadays, readers tend to consume news content on smaller user interface types (e.g., mobile devices) [10, 20], the impact of popularity bias may even get amplified due to the reduced number of recommendations that can be shown [12].

In this paper, we therefore discuss the introduction of personalized, content-based news articles on DiePresse, a popular Austrian news platform, focusing on two aspects: (i) user interface type, and (ii) popularity bias mitigation. To do so, we performed a two-weeks online study that started in October 2020, in which we compared the impact of recommendations with respect to different user groups, i.e., anonymous (cold-start [18]) and subscribed (logged-in and paying) users, as well as different user interface types, i.e., desktop, mobile and tablet devices (see Sect. 2). Specifically, we address two research questions:

RQ1: How does the user interface type impact the performance of news recommendations?

RQ2: Can we mitigate popularity bias by introducing personalized, content-based news recommendations?

We investigate RQ1 in Sect. 3.1 and RQ2 in Sect. 3.2. Additionally, we discuss the impact of two significant events, i.e., (i) the COVID-19 lockdown announcement in Austria, and (ii) the Vienna terror attack, on the consumption behavior of users. We hope that our findings will help other news platform providers assessing the impact of introducing personalized recommendations.

2 Experimental Setup

In order to answer our two research questions, we performed a two-weeks online user study, which started on the 27th of October 2020 and ended on the 9th of November 2020. Here, we focused on three user interface types, i.e., desktop, mobile and tablet devices, as well as investigated two user groups, i.e., anonymous and subscribed users. About 89% of the traffic (i.e., 2,371,451 user interactions) was produced by the 1,182,912 anonymous users, where a majority of them (i.e., 77.3%) read news articles on a mobile device. Interestingly, the 15,910 subscribed users exhibited a more focused reading behavior and only interacted with a small subset of all articles that were read during our online study (i.e., around 18.7% out of 17,372 articles). Within the two-weeks period, two significant events happened: (i) the COVID-19 lockdown announcement in Austria on the 31st of October 2020, and (ii) the Vienna terror attack on the 2nd of November 2020. The articles related to these events were the most popular ones in our study.

Calculation of Recommendations. We follow a content-based approach to recommend news articles to users [19]. Therefore, we represent each news article using a 25-dimensional topic vector calculated using Latent Dirichlet Allocation (LDA) [3]. Each user was also represented by a 25-dimensional topic vector, where the user's topic weights are calculated as the mean of the news articles' topic weights read by the user. In case of subscribed users, the read articles consist of the entire user history and in case of anonymous users, the read articles consist of the articles read in the current session. Next, these topic vectors are used to match users and news articles using Cosine similarity in order to find top- n news article recommendations for a given user. For our study, we set $n = 6$ recommended articles. For this step, only news articles are taken into account that have been published within the last 48 h. Additionally, editors had the possibility to also include older (but relevant) articles into this recommendation pool (e.g., a more general article describing COVID-19 measurements).

In total, we experimented with four variants of our content-based recommendation approach: (i) recommendations only including articles of the last 48 h, (ii) recommendations also including the editors' choices, and (iii) and (iv) recommendations, where we also included a collaborative component by mixing the user's topic vector with the topic vectors of similar users for the variants (i) and (ii), respectively. Additionally, we also tested a most-popular approach, since this algorithm was already present in DiePresse before the user study started. However, we did not find any significant differences between these five approaches with respect to recommendation accuracy in our two-weeks study and therefore, we did not distinguish between the approaches and report the results for all calculated recommendations in the remainder of this paper.

3 Results

3.1 RQ1: User Interface Type

Most studies focus on improving the accuracy of the recommendation algorithms, but recent research has shown that this has only a partial effect on the final user experience [13]. The user interface is namely a key factor that impacts the usability, acceptance and selection behavior within a recommender system [6]. Additionally, in news platforms, we can see a trend that shifts from classical desktop devices to mobile ones. Moreover, users are biased towards clicking on higher ranked results (i.e., position bias) [4]. When evaluating personalized news recommendations, it becomes even more important to understand the user acceptance of recommendations for smaller user interface types, where it is much harder for the user to see all recommended options due to the limited size. In our study, we therefore investigate to what extent the user interface type impacts the performance of news recommendations (RQ1). As mentioned, we differentiate between three different user interface types, i.e., interacting with articles on a (i) desktop, (ii) mobile, and (iii) tablet device. In order to measure the acceptance of recommendations shown via the chosen user interface type, we use the following two evaluation metrics [9]:

Table 1. RQ1: Acceptance of recommended articles with respect to user interface type.

Metric	Desktop	Mobile	Tablet
RSR: Recommendation-Seen-Ratio (%)	26.88	17.55	26.71
CTR: Click-Through-Rate (%)	10.53	13.40	11.37

Recommendation-Seen-Ratio (RSR) is defined as the ratio between the number of times the user actually saw recommendations (i.e., scrolled to the corresponding recommendation section in the user interface) and the number of recommendations that were generated for a user.

Click-Through-Rate (CTR) is measured by the ratio between the number of actually clicked recommendations and the number of seen recommendations.

As shown in Table 1, the smaller user interface size of a mobile device heavily impacts the probability of a user to actually see the list of recommended articles. This may be due to the fact that reaching the position where the recommendations are displayed is harder in comparison to a larger desktop or tablet device, where the recommendation section can be reached without scrolling. Interestingly enough, once a user has seen the list of recommended articles, users who use a mobile device exhibit a much higher CTR. Again, we hypothesize that if a user has put more effort into reaching the list of recommended articles, the user is more likely to accept the recommendation and interact with it.

When looking at Fig. 1, we can see a consistent trend during the two weeks of our study regarding the user interface types for both the RSR and CTR measures. However, notable differences are the fluctuations of the evaluation measures for the two significant events that happened during the study period. For instance, the positive peak in the RSR and the negative peak in CTR that can be spotted around the 31st of October was caused by the COVID-19 lockdown announcement in Austria. For the smaller user interfaces (i.e., mobile and tablet devices) this actually increased the likelihood of the recommendation to be seen since users have invested more energy in engaging with the content of the news articles. On the contrary, we saw a drop in the CTR, which was mostly caused by anonymous users since the content-based, personalized recommendations did not provide articles that they expected at that moment (i.e., popular ones solely related to the event). Another key event can be spotted on the 2nd of November, the day the Vienna terror attack happened. This was by far the most read article with a lot of attack-specific information during the period of the online study. Across all three user interface types, this has caused a drop in the likelihood of a recommendation to be seen at all. Interestingly enough, the CTR in this case does not seem to be influenced. We investigated this in more detail and noticed that a smaller drop was only noticeable for the relatively small number of subscribed users using a mobile device and thus, this does not influence the results shown in Fig. 1. The differences between all interface types shown in Table 1 and Fig. 1 are statistically significant according to a Kruskal-Wallis followed by a Dunn test except for mobile vs. tablet device with respect to CTR.

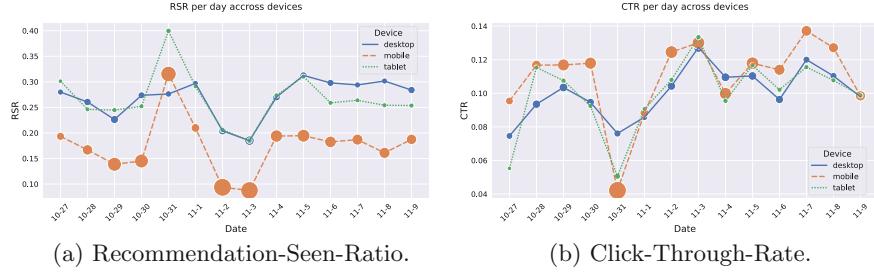


Fig. 1. RQ1: Acceptance of recommended articles for the two weeks of our study with respect to (a) RSR, and (b) CTR. The size of the dots represent the number of reading events on a specific day for a specific user interface type.

3.2 RQ2: Mitigating Popularity Bias

Many recommender systems are affected by popularity bias, which leads to an overrepresentation of popular items in the recommendation lists. One potential issue of this is that unpopular items (i.e., so-called long-tail items) are recommended rarely [15, 16]. The news article domain is an example where ignoring popularity bias could have a significant societal effect. For example, a potentially controversial news article could easily impose a narrow ideology to a large population of readers [7]. This effect could even be strengthened by providing unpersonalized, most-popular news recommendations as it is currently done by many online news platforms (including DiePresse) since these popularity-based approaches are easy to implement and also provide good offline recommendation performance [9, 10]. We hypothesize that the introduction of personalized, content-based recommendations (see Sect. 2) could lead to more balanced recommendation lists in contrast to most-popular recommendations. This way also long-tail news articles are recommended and thus, popularity bias could be mitigated. Additionally, we believe that this effect differs between different user groups and thus, we distinguish between anonymous and subscribed users.

We measure popularity bias in news article consumption by means of the skewness [2] of the article popularity distribution, i.e., the distribution of the number of reads per article. Skewness measures the asymmetry of a probability distribution, and thus a high, positive skewness value depicts a right-tailed distribution, which indicates biased news consumption with respect to article popularity. On the contrary, a small skewness value depicts a more balanced popularity distribution with respect to head and tail, and thus indicates that also non-popular articles are read. As another measure, we calculate the kurtosis of the popularity distribution, which measures the “tailedness” of a distribution. Again, higher values indicate a higher tendency for popularity bias. For both metrics, we hypothesize that the values at the end of our two-weeks study are smaller than at the beginning, which would indicate that the personalized recommendations helped to mitigate popularity bias.

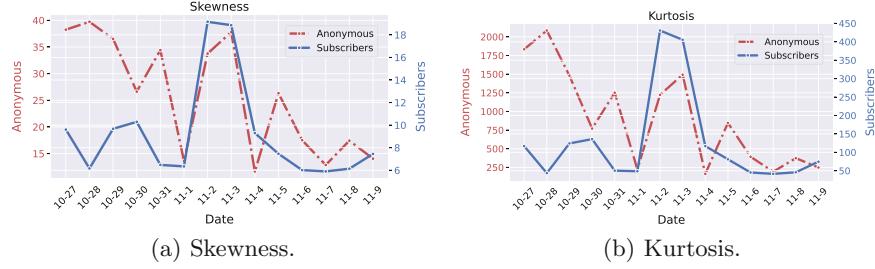


Fig. 2. RQ2: Impact of personalized, content-based recommendations on the popularity bias in news article consumption measured by (a) skewness and (b) kurtosis based on the number of article reads for each day.

The plots in Fig. 2 show the results addressing RQ2. For both metrics, i.e., skewness and kurtosis, we see a large gap between anonymous users and subscribers at the beginning of the study (i.e., 27th of October 2020), where only most-popular recommendations were shown to the users. While anonymous users have mainly read popular articles, subscribers were also interested in unpopular articles. This makes sense since subscribed users typically visit news portals for consuming articles within their area of interest, which will also include articles from the long-tail, while anonymous users typically visit news portals for getting a quick overview of recent events, which will mainly include popular articles. Based on this, a most-popular recommendation approach does not impact subscribers as much as it impacts anonymous users.

However, when looking at the last day of the study (i.e., 9th of November 2020), there is a considerably lower difference between anonymous and subscribed users anymore. We also see that the values at the beginning and at the end of the study are nearly the same in case of subscribed users, which shows that these users are not prone to popularity bias, and thus also personalized recommendations do not affect their reading behavior in this respect. With respect to RQ2, we find that the introduction of personalized recommendations can help to mitigate popularity bias in case of anonymous users. Furthermore, we see two significant peaks in the distributions that are in line with the COVID-19 lockdown announcement in Austria and the Vienna terror attack. Hence, in case of significant events also subscribed users are prone to popularity bias.

4 Conclusion

In this paper, we discussed the introduction of personalized, content-based news recommendations on DiePresse, a popular Austrian news platform, focusing on two specific aspects: user interface type (RQ1), and popularity bias mitigation (RQ2). With respect to RQ1, we find that the probability of recommendations to be seen is the highest for desktop devices, while the probability of clicking the recommendations is the highest for mobile devices. With respect to RQ2, we find

that personalized, content-based news recommendations result in a more balanced distribution of news articles' readership popularity for anonymous users. For future work, we plan to conduct a longer study, in which we also want to study the impact of different recommendation algorithms (e.g., use BERT [5] instead of LDA and include collaborative filtering) on converting anonymous users into paying subscribers. Furthermore, we plan to investigate other evaluation metrics, such as recommendation diversity, serendipity and novelty.

Acknowledgements. This work was funded by the H2020 projects TRUSTS (GA: 871481), TRIPLE (GA: 863420), and the FFG COMET program. The authors want to thank Aliz Budapest for supporting the study execution.

References

1. Abdollahpouri, H.: Popularity bias in ranking and recommendation. In: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, pp. 529–530 (2019)
2. Bellogín, A., Castells, P., Cantador, I.: Statistical biases in information retrieval metrics for recommender systems. *Inf. Retrieval J.* **20**(6), 606–634 (2017)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
4. Craswell, N., Zoeter, O., Taylor, M., Ramsey, B.: An experimental comparison of click position-bias models. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, pp. 87–94 (2008)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
6. Felfernig, A., Burke, R., Pu, P.: Preface to the special issue on user interfaces for recommender systems. *User Modeling User-Adapted Interact.* **22**(4–5), 313 (2012)
7. Flaxman, S., Goel, S., Rao, J.M.: Filter bubbles, echo chambers, and online news consumption. *Public Opinion Quart.* **80**(S1), 298–320 (2016)
8. Garcin, F., Faltings, B.: Pen recsys: a personalized news recommender systems framework. In: Proceedings of the 2013 International News Recommender Systems Workshop and Challenge, NRS 2013, pp. 3–9. Association for Computing Machinery, New York (2013). <https://doi.org/10.1145/2516641.2516642>
9. Garcin, F., Faltings, B., Donatsch, O., Alazzawi, A., Bruttin, C., Huber, A.: Offline and online evaluation of news recommender systems at swissinfo.ch. In: Proceedings of the 8th ACM Conference on Recommender Systems, RecSys 2014, pp. 169–176 (2014). <https://doi.org/10.1145/2645710.2645745>
10. Karimi, M., Jannach, D., Jugovac, M.: News recommender systems – survey and roads ahead. *Inf. Process. Manage.* **54**(6), 1203–1227 (2018). <https://doi.org/10.1016/j.ipm.2018.04.008>. <https://www.sciencedirect.com/science/article/pii/S030645731730153X>
11. Kille, B., et al.: Overview of clef newsreel 2015: News recommendation evaluation lab. In: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum (2015)
12. Kim, J., Thomas, P., Sankaranarayana, R., Gedeon, T., Yoon, H.J.: Eye-tracking analysis of user behavior and performance in web search on large and small screens. *J. Assoc. Inf. Sci. Technol.* **66**(3), 526–544 (2015)

13. Knijnenburg, B.P., Willemse, M.C., Gantner, Z., Soncu, H., Newell, C.: Explaining the user experience of recommender systems. *User Modeling User-Adapted Interac.* **22**(4), 441–504 (2012)
14. Kowald, D., Dennerlein, S.M., Theiler, D., Walk, S., Trattner, C.: The social semantic server a framework to provide services on social semantic network data. In: Proceedings of the I-SEMANTICS 2013 Posters & Demonstrations Track co-located with 9th International Conference on Semantic Systems (I-SEMANTICS 2013), pp. 50–54 (2013)
15. Kowald, D., Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E.: Support the underground: characteristics of beyond-mainstream music listeners. *EPJ Data Sci.* **10**(1), 1–26 (2021). <https://doi.org/10.1140/epjds/s13688-021-00268-9>
16. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: a reproducibility study. *Adv. Inf. Retrieval* **12036**, 35 (2020)
17. Lacic, E., Kowald, D., Seitlinger, P., Trattner, C., Parra, D.: Recommending items in social tagging systems using tag and time information. In: Proceedings of the 1st International Workshop on Social Personalisation (SP’2014) co-located with the 25th ACM Conference on Hypertext and Social Media (2014)
18. Lacic, E., Kowald, D., Traub, M., Luzhnica, G., Simon, J.P., Lex, E.: Tackling cold-start users in recommender systems with indoor positioning systems. In: Poster Proceedings of the 9th ACM Conference on Recommender Systems. Association of Computing Machinery (2015)
19. Lops, P., De Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Recommender Systems Handbook, pp. 73–105 (2011)
20. Newman, N., Fletcher, R., Levy, D., Nielsen, R.K.: The Reuters Institute digital news report 2016. Reuters Institute for the Study of Journalism (2015)
21. Seitlinger, P., Kowald, D., Kopeinik, S., Hasani-Mavriqi, I., Lex, E., Ley, T.: Attention please! a hybrid resource recommender mimicking attention-interpretation dynamics. In: Proceedings of the 24th International Conference on World Wide Web, pp. 339–345 (2015)
22. de Souza Pereira Moreira, G., Ferreira, F., da Cunha, A.M.: News session-based recommendations using deep neural networks. In: Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems, DLRS 2018, pp. 15–23. Association for Computing Machinery, New York (2018). <https://doi.org/10.1145/3270323.3270328>

P16 A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations (2023)

Fairness and Popularity Bias in Recommender Systems

P16 Kowald, D.*, Mayr, G.*[,], Schedl, M., Lex, E. (2023). A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations. In *Advances in Bias and Fairness in Information Retrieval (BIAS @ ECIR'2023)*. Communications in Computer and Information Science, vol. 1840, pp. 1-16. (*equal contribution)

DOI: https://doi.org/10.1007/978-3-031-37249-0_1



A Study on Accuracy, Miscalibration, and Popularity Bias in Recommendations

Dominik Kowald^{1,2}(), Gregor Mayr², Markus Schedl³, and Elisabeth Lex²

¹ Know-Center GmbH, Graz, Austria

dkowald@know-center.at

² Graz University of Technology, Graz, Austria

gregor.mayr@student.tugraz.at, {elisabeth.lex,dominik.kowald}@tugraz.at

³ Johannes Kepler University & Linz Institute of Technology, Linz, Austria

markus.schedl@jku.at

Abstract. Recent research has suggested different metrics to measure the inconsistency of recommendation performance, including the accuracy difference between user groups, miscalibration, and popularity lift. However, a study that relates miscalibration and popularity lift to recommendation accuracy across different user groups is still missing. Additionally, it is unclear if particular genres contribute to the emergence of inconsistency in recommendation performance across user groups. In this paper, we present an analysis of these three aspects of five well-known recommendation algorithms for user groups that differ in their preference for popular content. Additionally, we study how different genres affect the inconsistency of recommendation performance, and how this is aligned with the popularity of the genres. Using data from Last.fm, MovieLens, and MyAnimeList, we present two key findings. First, we find that users with little interest in popular content receive the worst recommendation accuracy, and that this is aligned with miscalibration and popularity lift. Second, our experiments show that particular genres contribute to a different extent to the inconsistency of recommendation performance, especially in terms of miscalibration in the case of the MyAnimeList dataset.

Keywords: Recommender systems · Popularity bias · Miscalibration · Accuracy · Recommendation inconsistency · Popularity lift

1 Introduction

Recommender systems benefit users by providing personalized suggestions of content such as movies or music. However, we also know from previous research that recommender systems suffer from an inconsistency in recommendation performance across different user groups [2, 9]. One example of this inconsistency is the varying recommendation accuracy across different user groups, which could

D. Kowald and G. Mayr—Both authors contributed equally to this work.

lead to unfair treatment of users whose preferences are not in the mainstream of a community [18, 19]. Other examples are inconsistencies between the input data of a recommender system and the recommendations generated, which could lead to recommendations that are either too popular and/or do not match the interests of specific user groups [2, 9]. Thus, popularity bias can be seen as one particular example of recommendation inconsistencies.

Apart from measuring recommendation accuracy differences across different user groups, related research [2] suggests quantifying the inconsistency of recommendation performance along two metrics, namely miscalibration and popularity lift. Miscalibration quantifies the deviation of a genre spectrum between user profiles and actual recommendations [24, 29]. For example, if a user listens to songs belonging to 45% pop, 35% rock, and 20% rap, whereas a calibrated recommendation list should contain the same genre distribution.

Related research also proposes the popularity lift metric to investigate to what extent recommendation algorithms amplify inconsistency in terms of popularity bias [3, 4]. This popularity lift metric quantifies the disproportionate amount of recommendations of more popular items in a system. For example, a positive popularity lift indicates that the items recommended are on average more popular than the ones in the user profile. Therefore, in the remainder of this paper, we refer to popularity lift as a metric that measures the popularity bias of recommendation algorithms.

However, a study that relates miscalibration and popularity lift to recommendation accuracy across different user groups is still missing. We believe that the outcomes of such a study could help choose the most suitable recommendation debiasing methods for each user group. Additionally, it is unclear if particular genres contribute to the emergence of inconsistency in recommendation performance across user groups. This knowledge could be helpful, e.g., for enhancing recommendation debiasing methods based on calibration.

The Present Work. In this paper, we contribute with a study on accuracy, miscalibration, and popularity bias of five well-known recommendation algorithms that predict the preference of users for items, i.e., UserItemAvg, UserKNN, UserKNNAvg [12], NMF [25], and Co-Clustering [10] in the domains of music (Last.fm), movies (MovieLens), and animes (MyAnimeList). We split the users in each dataset into three user groups based on the low, medium, and high inclination towards popular content, which we call LowPop, MedPop, and HighPop, respectively. With this, we aim to shed light on the connection between accuracy, miscalibration, and popularity bias in recommendations.

Furthermore, in this paper, we investigate what genres in the user groups are particularly affecting recommendation inconsistency across the algorithms and domains. With this, we aim to understand if particular genres contribute to the emergence of inconsistency in recommendation performance, and if this is aligned with the popularity of the genres.

Findings and Contributions. We find that LowPop consumers consistently receive the lowest recommendation accuracy, and in all investigated datasets,

miscalibration is the highest for this user group. In terms of popularity lift, we observe that all algorithms amplify popularity bias.

Concerning our analysis on the level of genres, we find that there are indeed genres that highly contribute to inconsistency, especially in terms of miscalibration in the case of the MyAnimeList dataset. In sum, the contributions of our paper are four-fold:

1. We extend three well-known datasets from the field of recommender systems with genre information to study the inconsistency of recommendation performance.
2. We evaluate five well-known recommendation algorithms for accuracy, miscalibration, and popularity lift.
3. We inspect recommendation inconsistency on the genre level and show that different genres contribute differently to the emergence of inconsistency in recommendation performance.
4. To foster the reproducibility of our work, we share the extended datasets and source code used in our study with the research community.

2 Related Work

Bias in information retrieval and recommender systems is an emerging research trait, and related works have shown multiple ways to quantify different biases in a system [7, 22]. One such bias is the popularity bias, which arises due to items with higher popularity getting recommended more often than items with lower popularity. Works [9] have found, that not all users are affected identically, with some user groups receiving more inconsistent recommendations than others. Ekstrand et al. [9, 17], for example, found inconsistencies in recommendation accuracy among demographic groups, with groups differing in gender and age showing statistically significant differences in effectiveness in multiple datasets. The authors evaluated different recommendation algorithms and identified varying degrees of utility effects.

Abdollahpouri et al. [2–4] also contributed to this line of research and introduced two metrics to quantify the inconsistency in recommendation performance from the user’s perspective. The first one is the miscalibration metric, which quantifies the misalignment between the genre spectrum found in a user profile and the genre spectrum found in this user’s recommendations. The second one is the popularity lift metric, which measures to what extent a user is affected by popularity bias, i.e., the unequal distribution of popular items in a user profile and this user’s recommendations. In datasets from the movie domain, they found that users that are more affected by popularity bias also receive more miscalibrated results. Similarly, Kowald et al. [19] analyzed popularity bias and accuracy differences across user groups in the music domain. The authors found that the popularity lift metric provided different results in the music domain than in the movie domain due to repeat consumption patterns prevalent in the music-listening behavior of users.

Table 1. Dataset statistics including the number of users $|U|$, items $|I|$, ratings $|R|$, and distinct genres $|C|$ as well as sparsity and rating range R -range.

Dataset	$ U $	$ I $	$ R $	$ C $	$ R / U $	$ R / I $	Sparsity	R -range
LFM	3,000	131,188	1,417,791	20	473	11	0.996	[1 – 1,000]
ML	3,000	3,667	675,610	18	225	184	0.938	[1 – 5]
MAL	3,000	9,450	649,814	44	216	69	0.977	[1 – 10]

In this paper, we extend these works by connecting miscalibration and popularity lift to recommendation accuracy across different user groups. Additionally, we examine if particular genres contribute to the emergence of recommendation inconsistency across user groups and datasets. With this, we hope to inform research on popularity bias mitigation methods. As an example, [5] has proposed in-processing methods for debiasing recommendations based on calibration. We believe that our findings on which genres contribute to miscalibrated results could be used to enhance these methods. Additionally, related research has proposed post-processing methods to re-rank recommendation lists [1,6]. We believe that our findings for the connection of accuracy and popularity lift for different user groups could help choose the right users for whom such re-ranking should be performed.

3 Method

In this section, we describe the datasets, the experimental setup, and the evaluation metrics used in our study.

3.1 Datasets

We use three different datasets in the domains of music, movies, and animes. Specifically, we use dataset samples from Last.fm (LFM), MovieLens (ML), and MyAnimeList (MAL) provided in our previous work [17]¹. Here, each dataset consists of exactly 3,000 users, which are split into three equally-sized groups with 1,000 users each. We use 1,000 users per user group to be comparable with previous works that also used groups of this size. The groups are created based on the users’ inclination toward popular items. Following the definitions given in [17], we define a user u ’s inclination towards popular items as the fraction of popular items in u ’s user profile. We define an item i as popular if it is within the top-20% of item popularity scores, i.e., the relative number of users who

¹ We do not use the BookCrossing dataset due to the lack of genre information.

have interacted with i . We term the group with the lowest, medium, and highest inclination toward popular items *LowPop*, *MedPop*, and *HighPop*, respectively. In Fig. 1, we show boxplots of the fraction of popular items in the user profiles of the three groups for our three datasets.

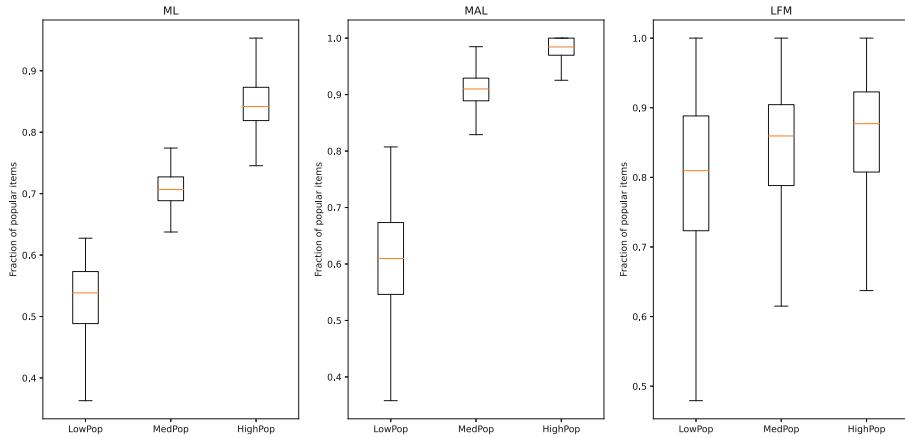


Fig. 1. Boxplots depicting the fraction of popular items in the user profiles for the three user groups and datasets. The LowPop group has the smallest ratio of popular items, compared to MedPop and HighPop. In the LFM dataset, this difference is not as apparent as in the other datasets, due to repeat consumption patterns in music listening behavior.

In Fig. 1, we show boxplots depicting the fraction of popular items in the user profiles for the three user groups and datasets. We see that the LowPop user group has the smallest ratio of popular items, compared to MedPop and HighPop. In the case of the LFM dataset, this difference is not as apparent as in the case of the other datasets, due to repeat consumption patterns in music listening behavior.

Basic statistics of the datasets can be found in Table 1, and we share our dataset samples via Zenodo². In the following, we give more details on these datasets and how we extend them with genre information. Additionally, we analyze the popularity distributions in the datasets on the levels of ratings and users to give context for our study on genre level, which follows later on.

Last.fm (LFM). The LFM dataset sample used in our study is based on the LFM-1b dataset [28] and the subset used in [17]. It contains listening records from the music streaming platform Last.fm. We include only listening records to music artists that contain genre information. Genre is acquired by indexing Last.fm's user-generated tags (assigned to artists) with the 20 main genres from

² <https://doi.org/10.5281/zenodo.7428435>.

the AllMusic database (top-3: rock, alternative, pop). When comparing the LFM dataset sample in Table 1 with the one from [17], we notice that the number of artists $|I|$ decreases from 352,805 to 131,188, which means that there is no genre information available in LFM for a large set of the long-tail artists. However, in terms of ratings, this leads to a relatively small reduction in ratings from 1,755,361 to 1,417,791. Following our previous work [17], we interpret the number of times a user has listened to an artist as a rating score, scaled to a range of [1; 1,000] using min-max normalization. We perform the normalization on the level of the individual user to ensure that all users share the same rating range, in which the user’s most listened artist has a rating score of 1,000 and the user’s least listened artist has a rating score of 1.

MovieLens (ML). Our ML dataset sample is based on the ML-1M dataset provided by the University of Minnesota [11]. Here, we gather the genre information for movies directly from the original dataset³, which provides genres for all movies and contains 18 distinct genres (top-3: comedy, drama, action). With respect to sparsity, ML is our densest dataset sample, while LFM is our sparsest one.

MyAnimeList (MAL). The MAL dataset used in our study is based on a recommender systems challenge dataset provided by Kaggle. As in the case of ML, the original dataset⁴ already provides genre information for each item, which leads to 44 distinct genres (top-3: comedy, action, romance). However, one special characteristic of MAL is that this dataset also contains implicit feedback (i.e., when a user bookmarks an anime). Following [17], we set the implicit feedback to an explicit rating of 5. In terms of the number of ratings, MAL is the smallest dataset used in our study, while LFM is the largest one.

Genre Popularity Distribution. To get a better understanding of the popularity of the individual genres across the three user groups, in Fig. 2, we plot the genre popularity distribution on the levels of ratings and users. The genres are ordered by their overall popularity in terms of ratings across all three user groups, i.e., the most popular genre is the leftmost. On the level of ratings (left plots), we see similar popularity distributions across all user groups. Interestingly, for ML and MAL, the LowPop group has the largest number of ratings across all genres, while for LFM this is the case for the MedPop group.

On the level of users, we identify similar popularity distributions across all user groups for LFM and ML. However, in the case of MAL, we see a prominent drop for the genre “Hentai” when investigating the MedPop and HighPop user groups. This is not the case for the LowPop user group, and thus, the preference for these genres among LowPop users exclusively could lead to an inconsistent recommendation performance for LowPop in the MAL dataset. When relating

³ <https://grouplens.org/datasets/movielens/1m/>.

⁴ <https://www.kaggle.com/CooperUnion/anime-recommendations-database>.

these results to the rating distributions on the left, we see no drop for the MedPop and HighPop user groups in the case of the “Hentai” genre. However, we see an increase in ratings for LowPop for this genre. This again shows the considerable interest of LowPop users for animes associated with the “Hentai” genre.

Finally, we also investigated the item popularity distributions across genres and user groups, where we did not inspect any noticeable differences when comparing the user groups on the genre level.

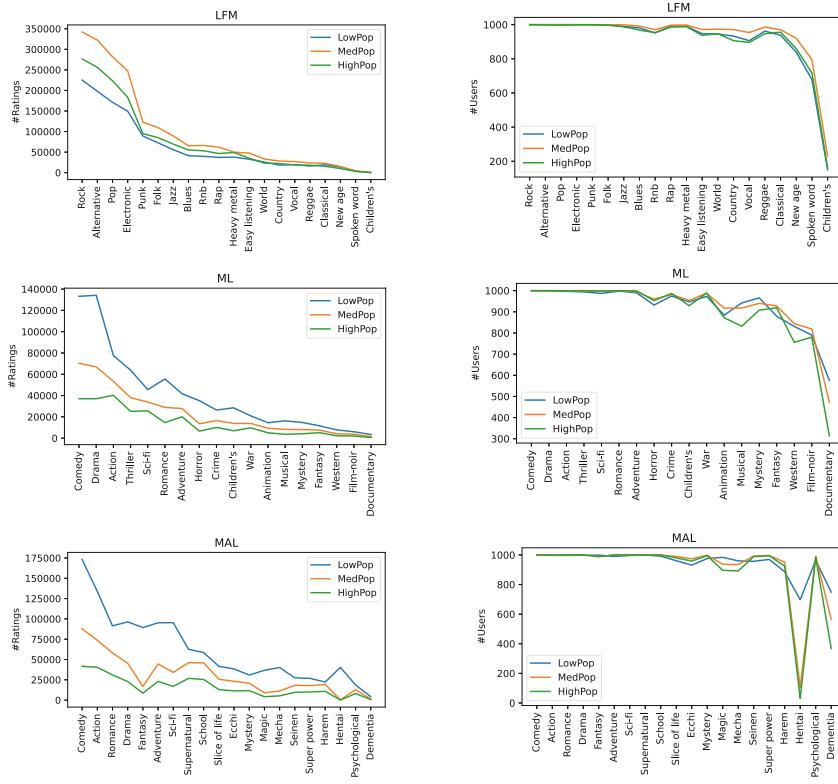


Fig. 2. Genre popularity distribution on the level of ratings (on the left) and on the level of users (on the right) for our three datasets and user groups.

3.2 Experimental Setup

Next, we describe the five recommendation algorithms and the evaluation protocol utilized in our study.

Recommendation Algorithms. Following our previous research [17, 19], we formulate the recommendation task as a rating prediction problem by utilizing

the Python-based Surprise framework [12]. Specifically, we use the four collaborative filtering (CF) recommendation algorithms studied in [17]. Since our previous work [17] also uses the same dataset samples as we do in the present work, we stick to the same hyperparameter settings. Please refer to our source-code shared via GitHub⁵ for the exact parameter settings. We refrain from performing any additional hyperparameter optimization since our main interest lies in assessing (relative) differences of our evaluation metrics between the three user groups LowPop, MedPop, and HighPop, and not in comparing a novel recommendation approach to state-of-the-art algorithms. This is also the reason why our focus lies on five traditional and easy understandable recommendation algorithms employed by related work instead of analyzing the performance of recent deep learning architectures, that would also lead to a much higher computational complexity.

The recommendation algorithms utilized in our study include the two KNN-based algorithms UserKNN and UserKNNAvg, where the latter one incorporates the average rating of the target user and item. We also study Co-Clustering, which is a scalable co-clustering-based CF approach [10], and NMF, i.e., non-negative matrix factorization [25]. Additionally, we add a non-CF approach utilized in [19], namely UserItemAvg, which predicts a baseline estimate using the overall average rating in the dataset and adds preference biases of the target user and item, e.g., if a user tends to give more positive ratings than the average user [14].

Evaluation Protocol. Concerning our evaluation protocol, we again follow our previous research [17, 19] and use a random 80/20 train/test split in a 5-fold cross-validation manner. Thus, we train our algorithms on the training set and measure the accuracy of the algorithms on the test set by comparing actual ratings with predicted ratings. By using 5-fold cross-validation, we ensure the robustness of our evaluation protocol, and control for potential fluctuations in the genre proportions or outliers in the recommendation calculations that may be introduced due to the random train/test splits.

For calculating miscalibration and popularity lift, we use a top-10 recommendation set for the target user, which are the 10 items with the highest predicted rating scores. Since our previous research [17, 19] has shown that the LowPop user group typically receives the worst recommendation accuracy across all user groups, we are especially interested in this user group. Therefore, we test for statistical significance using a t-test between LowPop and MedPop as well as between LowPop and HighPop. We report average values across all 5 folds for all metrics and indicate statistical significance only in case it applies for all 5 folds.

3.3 Evaluation Metrics

We quantify the inconsistency of recommendation performance using (i) accuracy differences between user groups, (ii) miscalibration, and (iii) popularity lift:

⁵ <https://github.com/domkowald/FairRecSys>.

Accuracy (MAE). We measure accuracy using the well-known mean absolute error (MAE) metric. The MAE of a user u is given by:

$$MAE(u) = \frac{1}{|R_u^{test}|} \sum_{r_{u,i} \in R_u^{test}} |r_{u,i} - R_{u,i}| \quad (1)$$

Here, the predicted rating score $R_{u,i}$ of user u and item i is compared to the real rating scores $r_{u,i}$ in u 's test set R_u^{test} . We favor MAE over the commonly used root mean squared error (RMSE) metric due to several disadvantages of RMSE, especially regarding the comparison of groups with different numbers of observations (i.e., ratings in our case) [30]. We report the MAE of a user group g by averaging the MAE values of all users of g .

To validate our accuracy results in terms of MAE also in top- n recommendation evaluation settings, we also report the well-known Precision and Recall metrics. For this, we classify an item in the test set as relevant if its rating is higher than the average rating in the train set.

Miscalibration (MC). The calibration metric proposed by Steck [29] quantifies the similarity of a genre spectrum between user profiles p and actual recommendations q . This metric was reinterpreted by Lin et al. [24] in the form of miscalibration, i.e., the deviation between p and q . We follow this definition and calculate the deviation using the Kullback-Leibler (KL) divergence between the distribution of genres in p , i.e., $p(c|u)$, and the distribution of genres in q , i.e., $q(c|u)$. This is given by:

$$KL(p||q) = \sum_{c \in C} p(c|u) \log \frac{p(c|u)}{q(c|u)} \quad (2)$$

Here, C is the set of all genres in a dataset. Therefore, $KL = 0$ means perfect calibration, and higher KL values (i.e., close to 1) mean miscalibrated recommendations. As in the case of MAE, we report the miscalibration values averaged over all users of a group g .

Popularity Lift (PL). The popularity lift metric investigates to what extent recommendation algorithms amplify the popularity bias inherent in the user profiles [3, 4]. Thus, it quantifies the disproportionate recommendation of more popular items for a given user group g (i.e., LowPop, MedPop, HighPop). We define the group average popularity $GAP_p(g)$ as the average popularity of the items in the user profiles p of group g . Similarly, $GAP_q(g)$ is the average popularity of the recommended items for all users of the group g . The popularity lift is then given by:

$$PL(g) = \frac{GAP_q(g) - GAP_p(g)}{GAP_p(g)} \quad (3)$$

Here, $PL(g) > 0$ means that the recommendations for g are too popular, $PL(g) < 0$ means that the recommendations for g are not popular enough, and $PL(g) = 0$ would be the ideal value.

4 Results

In this section, we describe and discuss the results of our study, first on a more general level and then on the level of genres.

Table 2. MAE, MC, and PL results for the LowPop, MedPop, and HighPop user groups. The highest (i.e., worst) results are highlighted in **bold**. Statistical significance according to a t-test between LowPop and MedPop, and LowPop and HighPop is indicated by * for $p < 0.05$. Rating ranges are shown in brackets.

<i>Algorithm</i>	<i>Data</i>	LFM [1-1,000]			ML [1-5]			MAL [1-10]		
		<i>Metric</i>	MAE	MC	PL	MAE	MC	PL	MAE	MC
UserItemAvg	<i>LowPop</i>	48.02*	0.52*	1.28	0.74*	0.78*	0.70*	0.99*	0.95*	1.12*
	<i>MedPop</i>	38.48	0.48	1.61	0.71	0.71	0.42	0.96	0.73	0.42
	<i>HighPop</i>	45.24	0.42	1.35	0.69	0.63	0.24	0.97	0.64	0.15
UserKNN	<i>LowPop</i>	54.32*	0.51*	0.52	0.80*	0.75*	0.64*	1.37*	0.92*	0.74*
	<i>MedPop</i>	46.76	0.50	0.82	0.75	0.69	0.37	1.34	0.72	0.22
	<i>HighPop</i>	49.75	0.45	0.80	0.72	0.62	0.20	1.31	0.63	0.08
UserKNNAvg	<i>LowPop</i>	50.12*	0.49*	0.35	0.76*	0.78*	0.49*	1.00*	0.90*	0.54*
	<i>MedPop</i>	40.30	0.47	0.61	0.73	0.70	0.33	0.95	0.73	0.24
	<i>HighPop</i>	46.39	0.42	0.64	0.70	0.61	0.20	0.95	0.64	0.11
NMF	<i>LowPop</i>	42.47*	0.54*	0.10	0.75*	0.78*	0.57*	1.01*	0.91*	0.87*
	<i>MedPop</i>	34.03	0.52	0.17	0.72	0.71	0.37	0.97	0.72	0.35
	<i>HighPop</i>	41.14	0.48	0.33	0.70	0.63	0.22	0.95	0.63	0.13
Co-Clustering	<i>LowPop</i>	52.60*	0.52*	0.68	0.74*	0.77*	0.70*	1.00*	0.90*	1.10*
	<i>MedPop</i>	40.83	0.51	1.04	0.71	0.70	0.43	0.96	0.72	0.42
	<i>HighPop</i>	47.03	0.45	0.99	0.68	0.62	0.25	0.98	0.63	0.16

Connection Between Accuracy, Miscalibration and Popularity Bias. Table 2 summarizes our results for the three metrics (MAE, MC, PL) over the three user groups (LowPop, MedPop, HighPop), three datasets (LFM, ML, MAL) and five algorithms (UserItemAvg, UserKNN, UserKNNAvg, NFM, Co-Clustering). The results presented are averaged over all users and all folds. We can see that in the case of ML and MAL, the LowPop user group receive the worst results for MAE, MC, and PL. These results are also statistically significant according to a t-test with $p < 0.05$. For LFM, the LowPop user group also gets the worst results for the MAE and MC metrics.

However, when looking at the PL metric, we observe different results, namely the highest popularity lift for either MedPop or HighPop. This is in line with

our previous research [19], which has shown that the PL metric provides different results for LFM than for ML. One potential difference between music and movies (and also animes) is that music is typically consumed repeatedly (i.e., a user listens to the same artist multiple times), while movies are mostly watched only once. The definition of the PL metric [24] does not account for repeat consumption patterns [15], since items are given the same importance regardless of their consumption frequency. This means that items that are consumed for instance 1,000 times by a specific user have the same importance as items that are consumed only once by this user.

Finally, in Table 3, we validate our accuracy results in terms of MAE also in top- n recommendation evaluation settings using the well-known Precision and Recall metrics. To classify relevant items in the test sets, we calculate the average rating in the training sets and treat a test item as relevant if it exceeds this average train rating. We see very similar results as in the case of the MAE metric. This means that in almost all cases, LowPop gets the worst results (i.e., lowest) and HighPop gets the best results (i.e., highest).

Table 3. Accuracy results in terms of Precision and Recall. We tested for statistical significance using a t-test between LowPop and MedPop, and LowPop and HighPop users, which is indicated by * for $p < 0.05$. The best (i.e., highest) results are highlighted in **bold**. The results are in line with the MAE ones, which means that LowPop receives worst accuracy results, while HighPop receives the best accuracy results.

Algorithm	Metric	LFM		ML		MAL	
		Precision	Recall	Precision	Recall	Precision	Recall
UserItemAvg	<i>LowPop</i>	0.30	0.11	0.78*	0.19*	0.71*	0.15*
	<i>MedPop</i>	0.28	0.08	0.82	0.26	0.80	0.21
	<i>HighPop</i>	0.39	0.14	0.83	0.36	0.80	0.33
UserKNN	<i>LowPop</i>	0.33*	0.16	0.78*	0.18*	0.71*	0.15*
	<i>MedPop</i>	0.38	0.14	0.83	0.25	0.80	0.22
	<i>HighPop</i>	0.53	0.22	0.83	0.35	0.81	0.34
UserKNNAvg	<i>LowPop</i>	0.34	0.16	0.80*	0.20*	0.73*	0.16*
	<i>MedPop</i>	0.34	0.12	0.83	0.27	0.80	0.23
	<i>HighPop</i>	0.47	0.19	0.83	0.36	0.81	0.36
NMF	<i>LowPop</i>	0.34	0.16	0.70*	0.14*	0.67*	0.13*
	<i>MedPop</i>	0.34	0.12	0.79	0.23	0.79	0.21
	<i>HighPop</i>	0.46	0.19	0.82	0.34	0.81	0.33
Co-Clustering	<i>LowPop</i>	0.33	0.16*	0.76*	0.17*	0.69*	0.14*
	<i>MedPop</i>	0.33	0.12	0.83	0.25	0.80	0.22
	<i>HighPop</i>	0.46	0.20	0.84	0.35	0.81	0.34

Influence of Genres on Inconsistency of Recommendations. Furthermore, Fig. 3 visualizes the results of our investigation on what genres in the user groups are particularly affecting inconsistency of recommendation performance in terms of miscalibration for the three datasets. We investigate this study for the miscalibration metric only, since we do not observe any particular differences across the genres for the MAE and popularity lift metrics. To map the users' miscalibration scores to a genre g , we assign the MC score of a user u to all genres listened to u . Then for each genre g , we calculate the average MC scores of all users of a specific user group who listened to g . These values are then plotted in Fig. 3 for both the NMF algorithm and the Co-Clustering algorithm. For better readability, we apply min-max normalization in a range of 0 - 1.

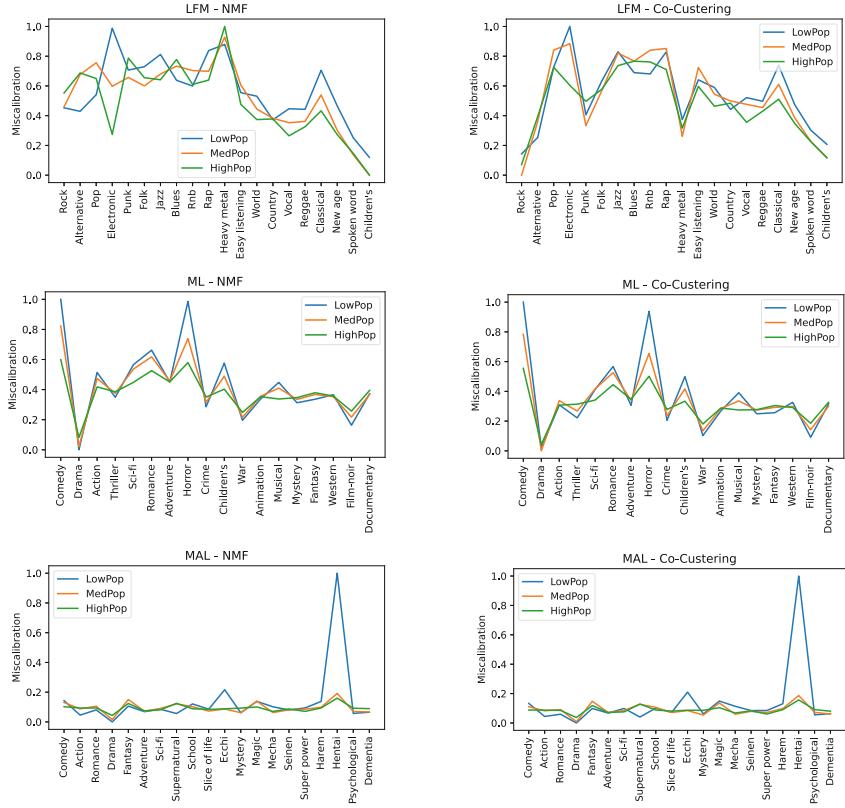


Fig. 3. Influence of different genres on MC for the NMF algorithm (on the left) and Co-Clustering (on the right). We see that some genres highly contribute to inconsistency, especially in case of animes (MAL).

As in the case of Fig. 2, the genres are ordered by their popularity. For the sake of space, we only show the results for NMF and Co-Clustering, which are, in general, inline with results obtained for the other algorithms. However, our

GitHub repository also allows the inspection of the results for the other algorithms. Additionally, for MAL, we exclude 24 genres for which no substantial fluctuations are observed. This leads to 20 shown genres, as in the case of LFM.

For the MAL dataset and the LowPop group, we observe highly miscalibrated results for the “Hentai” genre. In particular, indicated by its position, “Hentai” is an unpopular genre for most of the MAL users. However, as also shown in Fig. 2, for users within the LowPop group (and only for this user group), it is a relevant genre that is underrepresented in their recommendation lists. This demonstrates that there are indeed particular genres that contribute to a large extent to recommendation inconsistency for specific user groups.

5 Conclusion and Future Work

In this paper, we have studied the interconnection between accuracy, miscalibration, and popularity bias for different user groups in three different domains. Here, we measured popularity bias in terms of popularity lift, a metric that compares the popularity of items in recommendation lists to the popularity of items in user profiles. Additionally, we investigated miscalibration, a metric that compares the genre spectrums in user profiles with the ones in recommendation lists. We find that, in general, the inconsistency of recommendations in terms of miscalibration and popularity lift is aligned with lower accuracy performance.

One exception to this is the popularity lift metric in the case of music recommendations; however, this result is in line with our previous work [19], in which repeat consumption settings have been studied. Additionally, we find that different genres contribute differently to miscalibration and popularity lift. That finding is particularly pronounced in the case of anime recommendations for LowPop users and for genres that are unpopular among other user groups. Another contribution of our work is that we publicly share our datasets and source code investigated in this study with the research community.

Limitations and Future Work. One limitation of our work is that we have focused solely on datasets from the multimedia/entertainment domains, namely music, movies, and animes. Although we have investigated domains with and without repeat consumption patterns, for future work, we plan to also study other domains with respect to accuracy, miscalibration, and popularity bias. This could include recommendations in online marketplaces [20] or recommendations in social networks [16] and will contribute to the generalizability of our findings. To further strengthen the generalizability of our work, we also plan to conduct further experiments with novel recommendation algorithms employing deep learning-based methods [22].

Another limitation of our work is that we have used MAE, Precision, and Recall as the only metrics to measure the accuracy of recommendations. In the future, we plan to extend this by also investigating ranking-based metrics such as nDCG [13, 21] as well as metrics that measure the novelty and diversity of

recommendations [8]. In this respect, we also plan to enhance our evaluation protocol and move from random train/test splits to temporal train/test splits [27]. Finally, we also plan to do experiments with a higher number of user groups with a smaller number of users per group (e.g., 10 groups with 300 users per group). With this, we aim to address a potential limitation with respect to having different popularity tendencies within a group.

As a general path for future work, we plan to build on the findings of this paper to develop strategies to overcome the inconsistency of recommendation performance across different user groups. For example, for particular genres where we find high miscalibration, we aim to research calibration-based debiasing approaches [5]. Another possibility to address popularity bias in the recommender system could be to build models based on concepts from psychology [23]. Finally, we plan to investigate novel metrics to measure popularity lift in repeat consumption settings, e.g., music recommendations. Here, we plan to either introduce a weighted variant of the metric or investigate alternative methods for converting implicit feedback (e.g., play counts) into explicit ratings [26].

Acknowledgements. This research was funded by the “DDAI” COMET Module within the COMET - Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. Additionally, this work was funded by the Austrian Science Fund (FWF): P33526 and DFH-23.

References

1. Abdollahpouri, H., Burke, R., Mobasher, B.: Managing popularity bias in recommender systems with personalized re-ranking. In: The Thirty-second International Flairs Conference (2019)
2. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The impact of popularity bias on fairness and calibration in recommendation. arXiv preprint [arXiv:1910.05755](https://arxiv.org/abs/1910.05755) (2019)
3. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity Bias in recommendation. arXiv preprint [arXiv:1907.13286](https://arxiv.org/abs/1907.13286) (2019)
4. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The connection between popularity bias, calibration, and fairness in recommendation. In: Fourteenth ACM Conference on Recommender Systems, pp. 726–731 (2020)
5. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B., Malthouse, E.: User-centered evaluation of popularity bias in recommender systems. In: Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, pp. 119–129 (2021)
6. Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans. Knowl. Data Eng. **24**(5), 896–911 (2011)
7. Baeza-Yates, R.: Bias in search and recommender systems. In: Fourteenth ACM Conference on Recommender Systems, p. 2 (2020)

8. Castells, P., Hurley, N.J., Vargas, S.: Novelty and diversity in recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) *Recommender Systems Handbook*, pp. 881–918. Springer, Boston, MA (2015). https://doi.org/10.1007/978-1-4899-7637-6_26
9. Ekstrand, M.D., et al.: All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In: Conference on Fairness, Accountability and Transparency, pp. 172–186. PMLR (2018)
10. George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: Fifth IEEE International Conference on Data Mining (ICDM2005), p. 4. IEEE (2005)
11. Harper, F.M., Konstan, J.A.: The MovieLens datasets: History and context. *ACM Trans. Interact. Intell. Syst.* **5**, 1–19 (2015)
12. Hug, N.: Surprise: a python library for recommender systems. *J. Open Source Soft.* **5**(52), 2174 (2020)
13. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst. (TOIS)* **20**(4), 422–446 (2002)
14. Koren, Y.: Factor in the neighbors: scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data (TKDD)* **4**(1), 1–24 (2010)
15. Kotzias, D., Lichman, M., Smyth, P.: Predicting consumption patterns with repeated and novel events. *IEEE Trans. Knowl. Data Eng.* **31**(2), 371–384 (2018)
16. Kowald, D., Dennerlein, S., Theiler, D., Walk, S., Trattner, C.: The social semantic server: a framework to provide services on social semantic network data. In: 9th International Conference on Semantic Systems, I-SEMANTICS 2013, pp. 50–54. CEUR (2013)
17. Kowald, D., Lacic, E.: Popularity Bias in collaborative filtering-based multimedia recommender systems. In: Boratto, L., Faralli, S., Marras, M., Stilo, G. (eds.) *Advances in Bias and Fairness in Information Retrieval. BIAS 2022. Communications in Computer and Information Science*, vol. 1610, pp. 1–11. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-09316-6_1
18. Kowald, D., Muellner, P., Zangerle, E., Bauer, C., Schedl, M., Lex, E.: Support the underground: characteristics of beyond-mainstream music listeners. *EPJ Data Sci.* **10**(1), 1–26 (2021). <https://doi.org/10.1140/epjds/s13688-021-00268-9>
19. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: a reproducibility study. In: Jose, J.M., et al. (eds.) *ECIR 2020. LNCS*, vol. 12036, pp. 35–42. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45442-5_5
20. Lacic, E., Kowald, D., Parra, D., Kahr, M., Trattner, C.: Towards a scalable social recommender engine for online marketplaces: the case of apache solr. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 817–822 (2014)
21. Lacic, E., Kowald, D., Traub, M., Luzhnica, G., Simon, J.P., Lex, E.: Tackling cold-start users in recommender systems with indoor positioning systems. In: Poster Proceedings of the 9th {ACM} Conference on Recommender Systems. ACM (2015)
22. Lesota, O., et al.: Analyzing item popularity bias of music recommender systems: are different genders equally affected? In: Proceedings of the 15th ACM Conference on Recommender Systems, pp. 601–606 (2021)
23. Lex, E., Kowald, D., Seitlinger, P., Tran, T.N.T., Felfernig, A., Schedl, M., et al.: Psychology-informed recommender systems. *Found. Trends® Inf. Retrieval* **15**(2), 134–242 (2021)

24. Lin, K., Sonboli, N., Mobasher, B., Burke, R.: Calibration in collaborative filtering recommender systems: a user-centered analysis. In: Proceedings of the 31st ACM Conference on Hypertext and Social Media, pp. 197–206. HT 2020, Association for Computing Machinery, New York, NY, USA (2020)
25. Luo, X., Zhou, M., Xia, Y., Zhu, Q.: An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Trans. Industr. Inf.* **10**(2), 1273–1284 (2014)
26. Pacula, M.: A matrix factorization algorithm for music recommendation using implicit user feedback. Maciej Pacula (2009)
27. Quadrana, M., Cremonesi, P., Jannach, D.: Sequence-aware recommender systems. *ACM Comput. Surv. (CSUR)* **51**(4), 1–36 (2018)
28. Schedl, M.: The LFM-1b dataset for music retrieval and recommendation. In: Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, pp. 103–110 (2016)
29. Steck, H.: Calibrated recommendations. In: Proceedings of the 12th ACM Conference on Recommender Systems, pp. 154–162. RecSys 2018, Association for Computing Machinery, New York, NY, USA (2018)
30. Willmott, C.J., Matsuura, K.: Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Res.* **30**(1), 79–82 (2005)

P17 Long-Term Dynamics of Fairness: Understanding the Impact of Data-Driven Targeted Help on Job Seekers (2023)

Fairness and Popularity Bias in Recommender Systems

[P17] Scher, S., Kopeinik, S., Truegler, A., **Kowald, D.** (2023). Long-Term Dynamics of Fairness: Understanding the Impact of Data-Driven Targeted Help on Job Seekers. *Nature Scientific Reports*, 13:1727.

DOI: <https://doi.org/10.1038/s41598-023-28874-9>



OPEN

Modelling the long-term fairness dynamics of data-driven targeted help on job seekers

Sebastian Scher^{1,✉}, Simone Kopeinik¹, Andreas Trügler^{1,2,3} & Dominik Kowald^{1,2}

The use of data-driven decision support by public agencies is becoming more widespread and already influences the allocation of public resources. This raises ethical concerns, as it has adversely affected minorities and historically discriminated groups. In this paper, we use an approach that combines statistics and data-driven approaches with dynamical modeling to assess long-term fairness effects of labor market interventions. Specifically, we develop and use a model to investigate the impact of decisions caused by a public employment authority that selectively supports job-seekers through targeted help. The selection of who receives what help is based on a data-driven intervention model that estimates an individual's chances of finding a job in a timely manner and rests upon data that describes a population in which skills relevant to the labor market are unevenly distributed between two groups (e.g., males and females). The intervention model has incomplete access to the individual's actual skills and can augment this with knowledge of the individual's group affiliation, thus using a protected attribute to increase predictive accuracy. We assess this intervention model's dynamics—especially fairness-related issues and trade-offs between different fairness goals—over time and compare it to an intervention model that does not use group affiliation as a predictive feature. We conclude that in order to quantify the trade-off correctly and to assess the long-term fairness effects of such a system in the real-world, careful modeling of the surrounding labor market is indispensable.

Data-driven methods for decision support—also known as data-informed decision support systems, AI-based decision support, or algorithmic decision-making—form useful technologies in many fields and get more and more widespread, not only in the private but also in the public sector^{1,2}. However, this also raises concerns among the general public, as AI-based systems are prone to replicate biases present in data and application design. For instance³, found that users attributed females receive a lower number of high-paid job-adds than similar male users. While the data used may be correct in its collection and historical representation, it often depicts outdated societal norms and values, capturing historical inequities and cultural biases⁴. When entering data-driven applications, the resulting discrimination adversely affects minorities and groups that have already been discriminated against and disadvantaged in the past and consequently creates a reinforcement loop⁵. Some applications—for example, social scoring by authorities—are considered so problematic that in the European Union there is a plan to prohibit them⁶.

The labor market is an area in which the use of AI-based decision support must be examined with particular care, as there is a long tradition of discrimination against social groups in the labor market, for instance, based on ethnicity⁷ or gender⁸, which in turn is reflected in data.

Public employment services (PES) support people in finding jobs and play a very important social role in many countries. In the European Union, access to free employment services is even a fundamental right (art. 29 EU fundamental rights charter). Recently, the Austrian Public Employment Service (AMS) has started to use an AI-based system that categorizes job-seekers into three groups according to individuals' low, moderate, or high prospects in the labor market. This categorization allows providing different types (qualities) of help depending on the individual's group affiliation. Also, predominantly supporting the group with moderately good prospects is considered most (cost) efficient, as their labor market prospects can be raised to an acceptable level with relatively little effort. For individuals in the low-prospect group, on the other hand, more effort would be needed to achieve the same outcome.

In the AMS system, prospects are informed by calculated probabilities (i.e., predictions) of people finding a job in the near future (i.e., the next couple of months)⁹. The prediction model is trained on employment data

¹Know-Center GmbH, Graz 8010, Austria. ²Institute of Interactive Systems and Data Science, Graz University of Technology, Graz 8010, Austria. ³Department of Geography and Regional Science, University of Graz, Graz 8010, Austria. ✉email: sscher@know-center.at

collected in the past and therefore exhibits a historical bias. Socio-cultural norms and prejudices are reflected and cause attributes of a person, such as gender and caretaking obligations, to be most influential on the predictive outcome. The fact that gender is used as a predictor, as well as sociopolitical issues related to the “efficient” distribution of public services, led to a broad public debate about the ethical implications of using the system^{10–12}.

The main political goal behind systems such as the AMS system is usually the wish to make public services—in this case employment services—more efficient. However, this could potentially have harmful and unintended consequences. Existing group inequalities could be reinforced by the systems interventions over time, or new inequalities might emerge. However, if and to what extent this happens in a particular setting is not easy to predict. Dynamic systems can often act in unintuitive ways, and the inclusion of regularly updated statistical models makes the system even harder to understand. It has been shown that a system that is fair in a static context can still produce unfair results in the long run if it is regularly updated and provides feedback to the environment¹³. Other results show that the situation of a system that is unfair in the beginning might also worsen if a specific privacy method is applied¹⁴. Despite these difficulties, non-discrimination and fairness are among the ethical and legal requirements for AI systems^{15,16}, and it is thus essential to find ways to assess fairness aspects also of labor market intervention systems that use statistics and AI. A known issue is that the stakeholders involved in the development cycle, even if sensitized to the concern, often lack experience, processes, and tools to manage the complex set of issues^{17,18}. Our paper is inspired by the ideas and problems of the AMS system, but it is not a study of this particular system. Instead, we focus on the general idea of such systems and their long-term effects.

This paper serves two main purposes: (1) we provide an introduction to the complexity of assessing the long-term fairness effects on the population if a public authority provides targeted help in the labor market, based on data-driven methods that include protected attributes (e.g., gender). Targeted help, for the purposes of this paper, means that individuals or groups of individuals receiving help are selectively chosen based on predetermined criteria. (2) We provide an answer to the question “How can we assess long-term fairness in a dynamical system such as a labor market?” For this, we develop and present an approach on how to actually assess such long-term fairness impacts in a dynamical system such as a labor market.

Additionally, our study means to highlight the benefits of quantitative modeling of the surrounding environment as an essential part of the assessment of causal long-term effects. We focus on the following main aspects:

- Trade-offs between different long-term fairness goals (e.g., reducing inequality between groups versus correctly assessing individuals’ labor market prospects) when a PES provides targeted aid to job-seekers.
- Impact of targeted aid- versus non-targeted aid- on the long-term fairness of public authority interventions.

To investigate these aspects, we propose a combination of dynamical numerical modeling and data-driven models. The model captures the principle dynamics of a labor market situation in which a public authority intervenes with targeted aid. It consists of a replenishing pool of job seekers, defined via a skill model, the labor market, in which job-seekers do or do not get jobs, and the PES, which intervenes and changes the skills of the job-seekers. All these parts are abstractions of the real world and are kept as simple as possible while still capturing the basic dynamics.

The individual skill model is defined as simply as possible while at the same time being sophisticated enough to account for inequalities and, optionally, either full or incomplete knowledge of an individual’s skills. Due to the lack of openly available empirical data, we use synthetic data in our study.

We assume a population of individuals, where the prospect on the labor market of each individual is controlled by the personal skill set of that individual, described by a set of independent skill features. Additionally, each individual belongs to one of two groups, described by a protected attribute (e.g., gender). The average skill level between the two groups is not the same but shows significant overlap. If an observer has knowledge about all skill features of an individual, they can accurately compute the total skill level of that individual, which means that knowledge about the protected attribute (i.e., which group the individual belongs to) does not yield any additional information with regard to the individual’s skill level.

We further assume that there is a public authority that helps individuals in improving their skills in the labor market, which we will call the Public Employment Service (PES). To support the improvement of the skills of an individual, the PES provides access to services that are selected according to the individual’s current prospects in the labor market. While the labor market (employers) has access to all skill features, the public authority, however, does not have access to all skill features but only to a subset. This assumption can be justified by the fact that in real life, while both employers and public authority will have some common information about the job-seekers (degrees, years of work experience, etc), employers will have both more resources and more branch-specific knowledge for evaluating applicants (e.g., exact degrees, specializations, soft skills, etc). In addition, it has knowledge about the group affiliation (i.e., protected attribute). Because the total skills are not evenly distributed across the two groups, the knowledge about an individual’s group affiliation combined with historical data gives probabilistic information on their real skills: if the individual belongs to the group that has on average higher skills, also the likelihood that this individual has high skills is larger than if it belongs to the other group, even if there is no other distinction in attributes. This concept has previously been discussed in economic research¹⁹. Using this additional information, however, has two potential problems: (i) the model is probabilistic, and thus, the resulting predictions are only accurate *on average*, and (ii) the model is based on a protected attribute—therefore, legal and/or ethical reasons might prohibit utilizing this information, as it results in different treatment solely based on the affiliation to a certain group given by the protected attribute.

In order to gain a better understanding of the implications different approaches might entail, we compare two prediction models that the PES could implement: one that uses the protected attribute and one that does

not. With this, we can study the trade-off between mitigating disparities in personal skills among the two groups and the aim of preventing misclassifications based on a protected attribute.

Additionally, we make assumptions about the labor market and encode these assumptions in a simple dynamical model. The model has a pool of job-seekers with an influx and an outflux. The PES provides targeted help to the current job-seekers. The targeting of the help is based on the skills of individuals and the historical model record of how long it took individuals with different skills to find a job. With this model, we consider different scenarios/approaches of the PES on how it distributes its limited resources across individuals with different assumed skills. Furthermore, we test different assumptions about the job market (e.g., biased and unbiased) and investigate how it affects the impact of targeted vs. non-targeted help. The model we develop and use in this study is “dynamic” on two different levels: first, as it is an agent-based model, it is dynamic at the level of individuals; second, as a consequence of the intervention of the PES, it is dynamic in the adaptation of average skill levels. The latter is similar to what in the economics literature would be referred to as transition paths between different policy states.

Related work

From a fundamental rights perspective, the issue of fairness in data-driven applications is discussed in several reports of the European Union Agency for Fundamental Rights (FRA)^{20–22}. There are different definitions of fairness, depending on context, application and world-view, which occasionally contradict each other. From a legal perspective, a major problem is that court decisions are highly tailored to specific circumstances, which contrasts with quantitative, generic measures of fairness²³. A wide variety of quantitative fairness metrics and debiasing algorithms have also been proposed in research, e.g.,^{24,25,26}, investigates what people perceive as most fair and find that demographic parity (a.k.a. statistical parity) receives the highest level of agreement in several cases presented. They argue against the common practice of optimizing AI-based decisions toward multiple fairness goals, but to select the most meaningful metric in terms of social context. In this paper, we take as a sample environment - and thus social context - the labor market and the long-term effects of tailored measures to support job seekers. This is inspired by the AI-based decision support system used by the Austrian PES (AMS) that caused a great level of public controversy and already has been the subject of previous studies. Lopez¹⁰ for instance, elaborates extensively on algorithmic details, as well as their underlying human-based decisions and their possible implications on the affected population. An emphasis is set on gender aspects and potential (intersectional) discrimination. Authors also remark on the lack of research with respect to whether and how to include gender as an attribute in such an algorithm. Allhutter et al.¹¹ takes an approach based on critical studies and fairness to discuss the “inherent politics of the AMS algorithm”. While in our paper we do not explicitly investigate the AMS algorithm, we contribute to this line of research by investigating the long-term impact of different intervention models on job seekers and, in particular, by exploring the algorithmic consideration of a protected attribute that distinguishes groups (e.g., gender). To this end, we introduce a dynamical modeling approach that complements previous research on long-term dynamics of fairness.

Long term dynamics of fairness. Liu et al.²⁷ introduces a formal, one-step feedback model to estimate the long-term impact of fairness constraints. It is presented by the example of a credit distribution scenario, but can, however, also be adapted to other domains given necessary domain knowledge. Mouzannar et al.²⁸ goes beyond this approach and introduces a formal, yet flexible model that allows the study of both economic utility and social equality as a consequence of fairness interventions. In our model, we follow a more dynamic modeling approach and investigate fairness according to other characteristics, going beyond the change in population mean, while our study is also more specific to labor market interventions. Instead of adopting a general approach, we develop a specific dynamical model that reflects our problem setting. Kannan et al.²⁹ discusses fairness in colleague admission and graduate hiring based on a two-stage model. The study concludes that under real-life conditions, two defined fairness goals (i.e., being admitted and hired independent of group membership) are unreasonable to achieve. The study clearly extends simple static models, however, does not intend to study the long-term impact of the fairness goals but rather aims to identify environmental variables that could allow the achievement of defined fairness goals. D’Amour et al.¹³ presents an extensive aspect in how results of long-term modelling may differ from static evaluation settings. In three simulation scenarios (i.e., loan allocation, college admission, and attention allocation) they show how simple agent-environment models evolve over time. Their results highlight the need to assess the fairness of algorithmic systems in continuous time steps.

We add to this line of research and introduce a more complex dynamic modeling approach that depicts a rather systemic viewpoint on data-driven decision-making implications. This results in more complex, quantitatively evaluated simulations that, however, still simplify the real-world setting. We also study the dilemma between individual and group fairness that has been commonly discussed in AI applications, particularly in regard to data-driven decision support³⁰.

Economics. We use the concept of “labor-market-models” in a way that is targeted toward the main aspects of our study. In economics, a number of different labor market models are used to study the supply and demand of labor (e.g.,³¹). In that context, our modeling approach can be seen as agent-based modeling, which is also widely adopted in economic research (see³² for a historical overview of agent-based modeling of labor markets). Chaturvedi et al.³³ builds an agent-based model of the labor market for research purposes, in which agents are individual persons. Discrimination in the labor market is a widely studied topic in economics. Seminal work was done by Ken Arrow, who studied discrimination in the labor market back in the 70s¹⁹. One explanation for discriminatory results Arrow gives is imperfect information, which we consider a variable in our model. Caine³⁴ gives an overview on the early work on labor market discrimination. In³⁵, the authors give an over-

view of theory and empirical evidence of racial discrimination in the labor market. A recent text-book on the topic is³⁶ (especially chapter 12). Finally³⁷, argues that individual fairness constraints are insufficient to remove racial inequality from the US labor market. They suggest a “dual labor market” that could solve this problem by applying a dynamical approach. They also argue that such further-reaching approaches will be more and more important if employment processes are continuously automatized. Similar to our work and to¹³, they abandon the concept of static fairness. However, our work focuses on the trade-offs between different long-term fairness goals, as described in the subsequent sections. Cohen et al.³⁸ studies the efficiency of recruiting practices from an employer’s point of view, including the incorporation of fairness constraints. The unique point of our study is the inclusion of a PES that uses a continuously updated data-driven decision model. To the best of our knowledge, this has not been done before.

Methods

Personal skill model and data generation. In our personal skill model, each individual has a personal skill set s_{real} that is composed of two independent skill features x_1 and x_2 .

$$s_{real}(x_1, x_2) \equiv \frac{1}{2}(x_1 + x_2) \quad (1)$$

We call it “real” because we will differentiate it from observed and from predicted/assumed skills later on. In addition, each individual has a binary protected attribute x_{pr} that can have values of 0 and 1. In reality, this could for example be female or male, but here it is used in an abstract way. Central here is that the definition of s_{real} does not explicitly contain x_{pr} .

We draw x_1 and x_2 from uncorrelated truncated normal distributions. This means there is no correlation between x_1 and x_2 . We use normal distributions because personal features such as “talent” are usually assumed to be normally distributed (e.g.,³⁹). Truncation is used to ensure that no one has x_1 and/or x_2 higher than the maximum reachable values in the intervention model (see Section “Intervention model” below). The distribution is truncated at plus-minus two times the standard deviation. Therefore, despite the truncation, the distribution is very similar to a non-truncated normal distribution, and thus appropriate for describing personal features. Furthermore, we assume that x_1 is completely independent of x_{pr} :

$$x_1 \in \mathcal{N}_{trunc}(0, 1), \quad (2)$$

The values of the binary protected attribute have equal probability:

$$x_{pr} \in \{0, 1\}, \quad p(x_{pr} = 0) = p(x_{pr} = 1) = \frac{1}{2} \quad (3)$$

In words, for generating our artificial population, we draw x_1 from a truncated normal distribution (truncated at x_{max}), and x_{pr} from the binary distribution $\{0, 1\}$ with uniform probability.

Thus, the probability of an individual belonging to a particular group (with respect to the protected attribute) is 50% for both groups, and both groups are therefore of equal or near equal size.

The second skill feature, x_2 , is correlated with x_{pr} and is generated with the following formula:

$$x_2 \in \frac{1}{2} \left\{ \alpha_{pr} \cdot \left(x_{pr} - \frac{1}{2} \right) + \mathcal{N}_{trunc}(0, 1) \right\} \quad (4)$$

The parameter α_{pr} controls how much x_2 is higher on average in the privileged group compared to the underprivileged group. The factor $\frac{1}{2}$ is subtracted from x_{pr} to ensure that x_2 has a mean of zero. When x_2 is generated this way, the individuals with $x_{pr} = 0$ have *on average* lower x_2 , and therefore on average lower s_{real} . To reflect this, we will from now on call the group of individuals with $x_{pr} = 0$ the *underprivileged group*, and individuals with $x_{pr} = 1$ the *privileged group*. Importantly, however, not all individuals in the underprivileged group have low x_2 and low s_{real} . There are individuals in the privileged group that have lower skills than some individuals in the underprivileged group, and there are individuals in the underprivileged group that have a skill that is above the population mean. The joint distribution of x_1 and x_2 and the distribution of s_{real} of a sample from the background population is shown in Fig. 1.

From the way x_1 and x_2 are generated and the fact that s_{real} per definition (Eq. 1)) can be completely inferred from x_1 and x_2 , follow two central facts: Given x_1 and x_2 , there is no additional information contained in x_{pr} when one wants to infer s_{real} (even though s_{real} is correlated with x_{pr}). If, however, one has only access to x_1 and at the same time, information on the distribution of s_{real} over the two groups (e.g., the mean of s_{real} separately for each group), and one wants to infer s_{real} , then including x_{pr} in addition to x_1 in a statistical prediction system yields additional information, even though s_{real} is completely defined by x_1 and x_2 . This will form the backbone of our study. If it is for example known that an individual has an average value of x_1 , but we do not know x_2 , then knowing x_{pr} will be decisive in estimating whether s_{real} of that person is below or above average: if the individual belongs to the underprivileged group, then the expectation would be that s_{real} is below average, but if the individual, with an unchanged value of x_1 , is in the privileged group, then the expectation would be that s_{real} is above average.

For our model, we assume that there is an (unlimited) *background-population* pool with the distribution of x_1, x_2, x_{pr} and s_{real} described by Eqs. (1)–(4). This background population and the distribution of the features of the individuals do not change throughout a model run, but acts as a pool for refilling the pool of job-seekers.

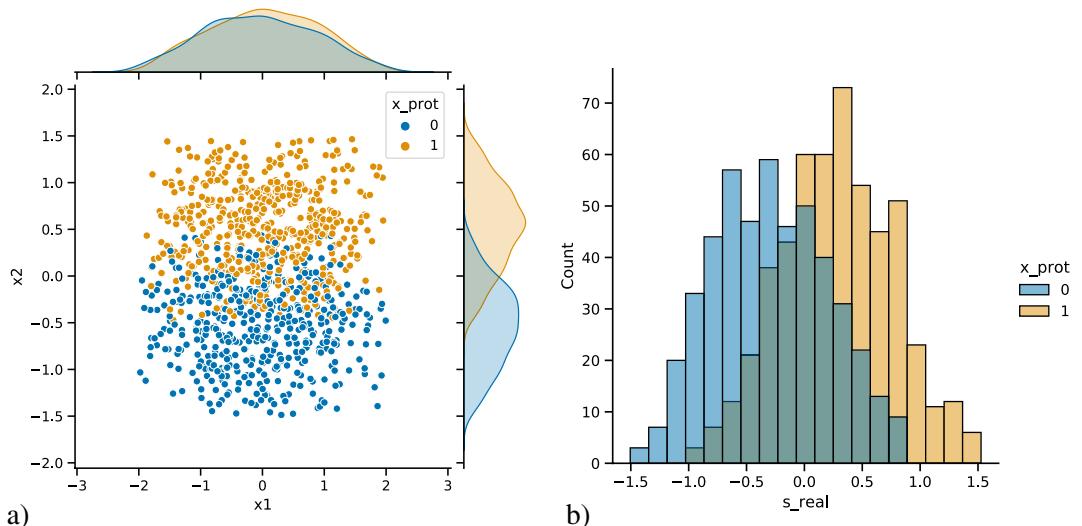


Figure 1. Sample of the background-population (a) Distribution of the two skill features, (b) Distribution of total skills (s_{real}), both split up according to the binary protected attribute. In (b), both colors are half-transparent, and the overlapping region is therefore depicted by the mixed color.

Prediction model. The prediction model is used by the PES in order to group individuals into a high-prospect and a low-prospect group, depending on the expected time-span T_u they will be unemployed (high-prospect = expected to find a job seen without help, low-prospect = expected to take longer to find a job without help). This is based on the time it took individuals to find a job in the past. The history is continuously build up throughout a model run with all individuals that found a job. The basis of this study is that the PES has access to an incomplete set of skill features only, namely solely to x_1 , and additionally access to x_{pr} . This assumption is reasonable because, in the real world, the PES will only have limited information about an individual (e.g. their education level and employment history), without access to more detailed information such as detailed CVs, job interviews, tests, etc. The simplest way to model this is through having 2 skill features, of which the PES can observe only one. To estimate (predict) the prospect group (above or below average s_{real}) from this, logistic regression is used to create the main (full) prediction model:

$$P_{\text{full}}(T_u > T_u^{\gamma} | x_1, x_{\text{pr}}) = \frac{1}{1 + e^{-(\alpha_1 x_1 + \alpha_2 x_{\text{pr}} + \beta)}} \quad (5)$$

Here, the parameters α_1, α_2 and β are estimated from the historical record, and T_u^{γ} is the threshold set on the unemployment time T_u for dividing the low and the high prospect group.

Additionally, we use a second prediction model, which we will call the *base model*, that does not use x_{pr} :

$$P_{\text{base}}(T_u > T_u^{\gamma} | x_1) = \frac{1}{1 + e^{-(\alpha^* x_1 + \beta^*)}} \quad (6)$$

with free parameters α^* and β^* fitted on the historical record. Logistic regression falls in a broader category of methods often referred to as supervised machine learning. Also, other supervised machine-learning algorithms, such as neural networks, could be used in the prediction model. Our choice for logistic regression was motivated by the fact that (i) it is the same method as used in the model that inspired our work i.e., the AMS-system, and (ii) it is a simple and easy to interpret method which allows for a certain level of transparency (in contrast to e.g., neural networks). A comparison of the full and the base model is necessary for investigating whether there are differences between different long-term fairness goals, as using the full prediction model conflicts with the fairness goal of not using the protected attribute but might be out-weighted by other long-term effects.

Labor market model. The labor market is modeled via a probabilistic function that for each individual defines the probability of finding a job at the current timestep, where this probability depends on s_{real} of that individual. We model the dependence on s_{real} as a logistic function:

$$P(\text{job} | s_{\text{real}}) = \frac{1}{1 + e^{-(\alpha_l s_{\text{real}} - \beta_l + b)}} \quad (7)$$

where α_l and β_l are fixed parameters. The parameter α_l controls the influence of s_{real} on the probability of finding a job, β_l sets at which value of $s_{\text{real}} + b$ the probability is 0.5. At each timestep, $P(\text{job} | s_{\text{real}})$ is computed for each individual within the pool of job-seekers. Each individual is removed from the pool of job seekers with a probability of P . The value b describes how biased the labor market is in favor of the privileged group. It is computed from a fixed labor market bias parameter β_b and the protected attribute:

$$b = \beta_b(x_{pr} - 0.5) \quad (8)$$

With $\beta_b = 0$ the labor market is unbiased, with $\beta_b > 0$ it is biased in favor of the privileged group. We use two different values in our experiments: 0 (“unbiased”), and 2 (“biased”).

The choice for a logistic labor market function was made because it satisfies the following intuition about the labor market: if an individual has very low skills, then the probability of finding a job is very low (close to zero), and if the skills slightly increase, then the chance is still very low. There is a soft threshold that one needs to reach in order to have a reasonable chance. Above this soft threshold, increases in s_{real} have a strong impact. Thus, the higher skills an individual has, the higher the chances of finding a job. Eventually, however, this reaches a plateau, as the probability of finding a job is already close to 1, and additional skills do basically not change anything anymore. The parameters α_l and β_l define this “middle” region, in which changes of s_{real} have a strong impact on the probability. β_l defines the position of this middle region, and α_l how broad/steep it is.

Note that we made $P(job|s_{real})$ independent of the time an individual is already unemployed. The intuition behind this is that in our idealized setting, the skills of an individual are solely defined by s_{real} , which the labor market knows. Therefore, in this setting, the fact that someone has been unemployed for a long time does not yield additional information about their skills. In reality, this might not necessarily be the case, since long-term unemployment as additional information could be a reason for an employer not to hire someone.

Intervention model. The intervention model describes the effect that the helping intervention of the PES has on the individual. The treatment of individuals differs between the high- and the low-prospect group in two ways: (i) in the amount of help (increase of x_1 and x_2) they receive, and (ii) in how long this help takes. The high-prospect group receives the help immediately and is available on the labor market in the next timestep. The low-prospect group, on the other hand, receives help that takes time and removes them from the labor market for a certain period of time ΔT_u . This is a simplified version of the current strategy of the Austrian PES (AMS).

The change in the individual skills features x_1 and x_2 depends on the current values, with decreasing increments as the skill features grow, approaching the limits set by the constants x_1^{max} and x_2^{max} :

$$x_1^{t+1} = \max[x_1^t + k(x_1^{max} - x_1^t), x_1^t] \quad (9)$$

$$x_2^{t+1} = \max[x_2^t + k(x_2^{max} - x_2^t), x_2^t] \quad (10)$$

The model parameter k defines how fast x_1 and x_2 grow. For simplicity, we use the same growth rate for both skill features. The choices for the value of k are described in Section “Scenarios”. Since individuals classified as low-prospect are removed from the active group for ΔT_u timesteps, their skills are updated $\Delta T_u + 1$ times (by applying Eqs. (9) and (10) $\Delta T_u + 1$ times) to account for that. Individuals that have been unemployed for too long (set by T_u^{max}) leave the system automatically.

As the model has random components, we run each simulation 10 times and average the results. All parameters of the model and their values are listed in Table B1 in the Supporting Information. The parameter values were set after testing different combinations. As we do not attempt to model an existing real-world setting, we have chosen a configuration that reaches reasonable equilibria for our main experiments, and the model with these parameters does not necessarily correspond to a specific use case. The sensitivity of the parameter choices is tested with additional experiments (see Supporting Information). An overview of the labor market and PES model is shown in Fig. 2.

Intervention scenarios. The value of k in the intervention model from Eqs (9) and (10) is central to our study, as it defines how strongly the intervention model affects different people. Testing different values is necessary to determine if there are differences between targeted and non-targeted help. To this end, we make k dependent both on the real prospect group C_r , and the prospect group C_{pr} predicted by the prediction model. The fact that the growth rate is made depended on the predicted prospect group reflects the idea of targeted help for different prospect groups, and that a prediction model is used to do this. Our idea is not only that different prospect groups receive a different *quantity* of help, but also a different *quality* that is better suited for that prospect group. Therefore, we also make k dependent on the real prospect group of each individual, as arguably if a specific type of help is better suited for the low then for the high prospect group, this will have the adverse effect for an incorrectly classified person.

The real prospect of a person cannot be precisely known, as the prospect is the *expected* time T_u that the individual will be unemployed. The real-time that this individual will need cannot be known, as it evolves from the model and is linked to s_{real} (but only in a *probabilistic* way). However, for the effects of the intervention model, we need something that is at least close to the real prospect. We, therefore, define the “real prospect” of an individual as T_u estimated from the historical data, using s_{real} as a predictor (in contrast to the predicted prospect, which uses x_1 and—if the full prediction model is used— x_{prot} as predictor). As for the predicted prospect, high and low prospect groups are defined via the same threshold T_u^Y , and each group is predicted with logistic regression.

Since both the real prospect group and the predicted prospect group are binary, this leads to a 2x2 matrix k_{ij} :

	predicted low	predicted high
real low	k_{11}	k_{12}
real high	k_{21}	k_{22}

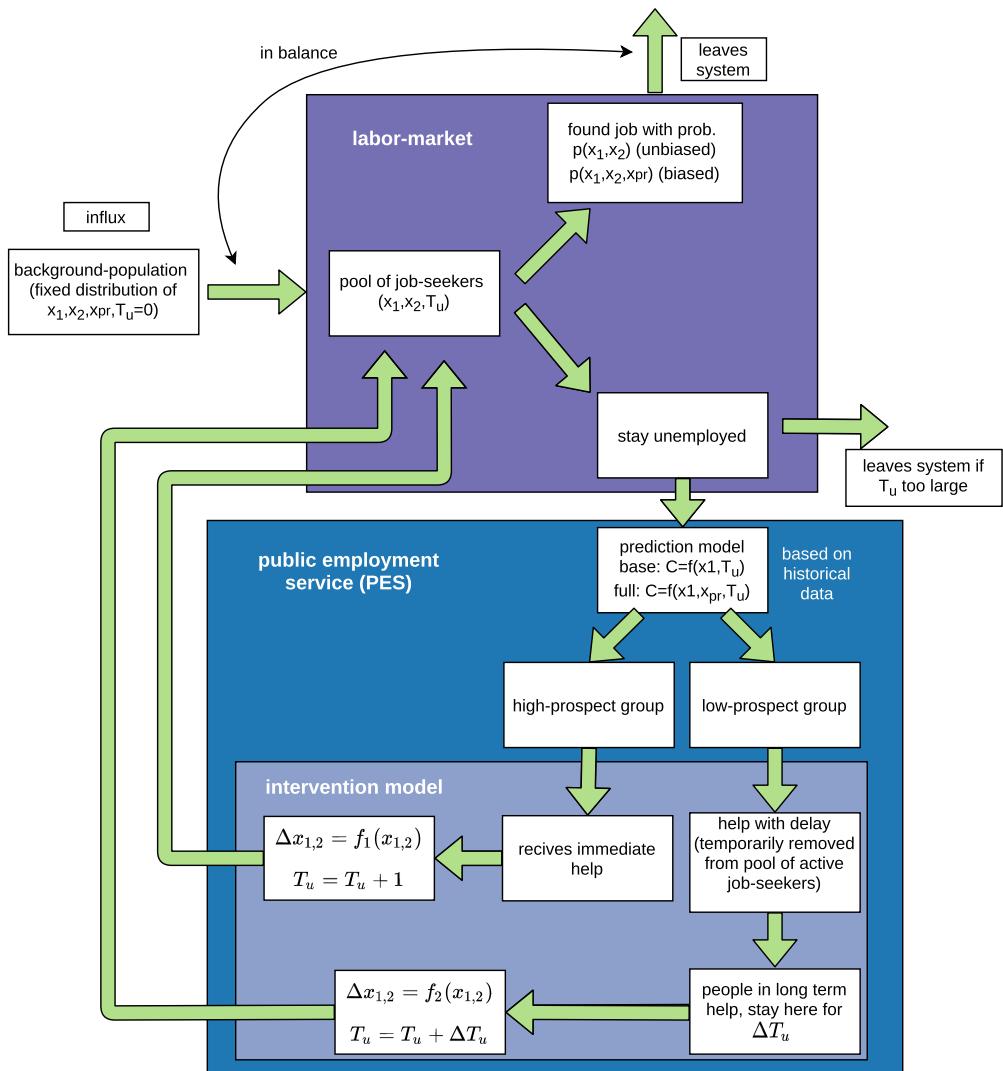


Figure 2. Outline of the labor market and PES model developed for this study. The labor market selects job-seekers from the pool of job seekers, with a probability dependent on the skills of the individual. Individuals who find a job leave the system, and individuals who have not found a job are transferred to the PES. The PES divides them into two groups, according to their predicted prospects in the labor market. The group with high prospects receives help (increase in skills) immediately and goes back to the pool of job seekers in the next timestep. The individuals in the group with low prospects receive help that takes more time and are withheld from the labor market for T_{delay} timesteps.

With different values for k_{ij} we can now define different settings, which we call *intervention scenarios*. The difference between k_{11} and k_{22} defines how different the effect of the intervention model is for the two different prospect groups, as intended by the PES. The differences between k_{11} and k_{21} and between k_{12} and k_{22} define how individuals are adversely affected if the prediction algorithm incorrectly classifies them. In this case, the groups receive the type of help that is intended for the other group.

The values for the different entries of k_{ij} define how—in the abstract setting of our model—“attention” or “resources” are distributed across the different groups.

For better readability, the k -values presented in the text and the plots are multiplied with a factor of 500, so for the k -matrix for scenario 1 all entries will be displayed as 1. We will use the following scenarios:

1. *Balanced*: $k_{11} = k_{12} = k_{21} = k_{22} = 1$. This is the base scenario, where all prospect groups receive the same quantity and quality of help, which also has the same effect, independent of the actual labor market prospects.
2. *Onlylow*: $k_{11} = k_{21} = 1, k_{12} = k_{22} = 0$. Only individuals classified as low-prospect receive help. The effect of the help is independent of whether the classification is correct or not.
3. *Onlyhigh*: $k_{11} = k_{21} = 0, k_{12} = k_{22} = 1$. Only individuals classified as high-prospect receive help. The effect of the help is independent of whether the classification is correct or not.

4. *Balanced_errors_penalized*: $k_{11} = k_{22} = k_{12} = 1, k_{21} = 1/2$. Both high and low-prospect groups receive the same amount of help, but if the classification is incorrect, the help is only half as effective.

Detailed descriptions of the scenarios are given in Table A1 in the “Supporting information”. The different scenarios show different targeting of the aid of the PES (no targeting in scenario 1, different ways of targeting in scenarios 2-4), and comparing them thus allows to answer what impact targeted aid vs. non-targeted aid has on the long-term fairness of public authority interventions.

Spin up phase. Each model run is started with a spin-up phase. The first 400 timesteps are run without the intervention model in order to allow the buildup of an initial historical dataset. The first 200 timesteps are discarded, and the remaining 200 are used as the initial historical dataset - the historical dataset that will be used in the first step of the model that includes the PES. With each further step of the model, the historical dataset will be extended. Thus, in the longer run, the historical set will more and more be influenced by the predictions made by the PES.

Metrics. In order to address trade-offs between different long-term fairness goals, we need to quantitatively define fairness for our setting. Our first metric is the Between Group skills Difference (*BGSD*), which is given by the difference of the mean skills between the two groups:

$$BGSD = \bar{s}_{real, x_{pr}=0} - \bar{s}_{real, x_{pr}=1} \quad (11)$$

This is a group fairness metric, and it is a property only of the data, not the prediction model. Our second metric is the fraction of the individuals that are predicted as low-prospect by the model but actually are high-prospect and would be classified as high-prospect by the model if the individuals would have the opposite protected attribute, which we call *counterfactual fraction*:

$$\frac{N(C_{p,x_{pr}} = \text{low} \wedge C_{p,x_{pr}^*} = \text{high} \wedge C_{tr} = \text{high})}{N(C_p = \text{low})} \quad (12)$$

Here, N is the number of individuals as a function of the respective prospect groups: C_p is the predicted prospect group, C_{tr} is the true prospect group, and x_{pr}^* is the opposite protected attribute of x_{pr} . Both metrics are computed at each timestep of the model. *BGSD* and *counterfactual fraction* represent different fairness goals, and a comparison is thus essential for addressing whether there are trade-offs between different long-term fairness goals.

As the third metric, we use Equal Opportunity, which is the difference in True Negative Rates (TNR) between the two groups:

$$EO = TNR_{priv} - TNR_{upriv} \quad (13)$$

where negative means predicted low prospect class. It is the fraction of low-prospect predictions that are really low prospect. Equal opportunity is a widely used fairness metric (e.g. ^{25,40}). Both counterfactual fraction and equal opportunity are properties not only of the data but also of the prediction model.

Results

For each of the four scenarios, the model was run with the base prediction model and full prediction model, and with either the unbiased or biased labor market, resulting in four model combinations. In order to get a better feel and intuition for the model, we start by looking at a single model run in detail. Then, all scenarios and model configurations are compared with respect to the *BGSD* and *counterfactual fraction* metrics.

Single model run (scenario: *onlylow*, model: *full prediction*). A run with full prediction (including the protected attribute) and unbiased labor market for scenario *onlylow* is shown in Fig. 3. The prediction-model performance metrics are only available for the time the prediction model is active. Shown is part of the spin-up phase (the first 200 timesteps were discarded) and at timestep 400 the PES intervention model kicks in. This is clearly seen in many of the shown measures. The skills s_{real} start to increase. For both the privileged and the underprivileged group, the mean skills increases, but it increases more for the underprivileged group, as can be seen by decreasing *BGSD*. At around timestep 500 the pool of job seekers reaches a new equilibrium in skills. The average time that the individuals, who found a job at a certain time-point were already unemployed (T_u), shows a more complex dynamic (panels in row two, which show T_u and between group difference in T_u (*BGTuD_current*)). Shortly after the PES starts its intervention, T_u increases for both groups and then decreases again. The fraction of underprivileged individuals in the current pool of job seekers (individuals looking for a job and individuals in the waiting group of the PES, *frac_uppriv* in the plot) is on the order of 0.8. The background population has—per how we generate our population data—a fraction of underprivileged individuals of 0.5. The reason that it is higher in the active group is that individuals in the privileged group have on average higher skills, and are thus more likely to find a job soon, leading to this imbalance in the active group. The plot *fraction_in_waiting* shows the fraction of individuals from the privileged group and the fraction of individuals from the underprivileged group in the job-seeker pool that are currently in the waiting position (where the low prospect individuals receive help while being withheld from the labor market). In the beginning, this fluctuates strongly. This fluctuation stems from the fact that in the first timestep after the PES starts its intervention, all low-prospect individuals from the current pool are put in the waiting group, and the pool is then filled up with random individuals

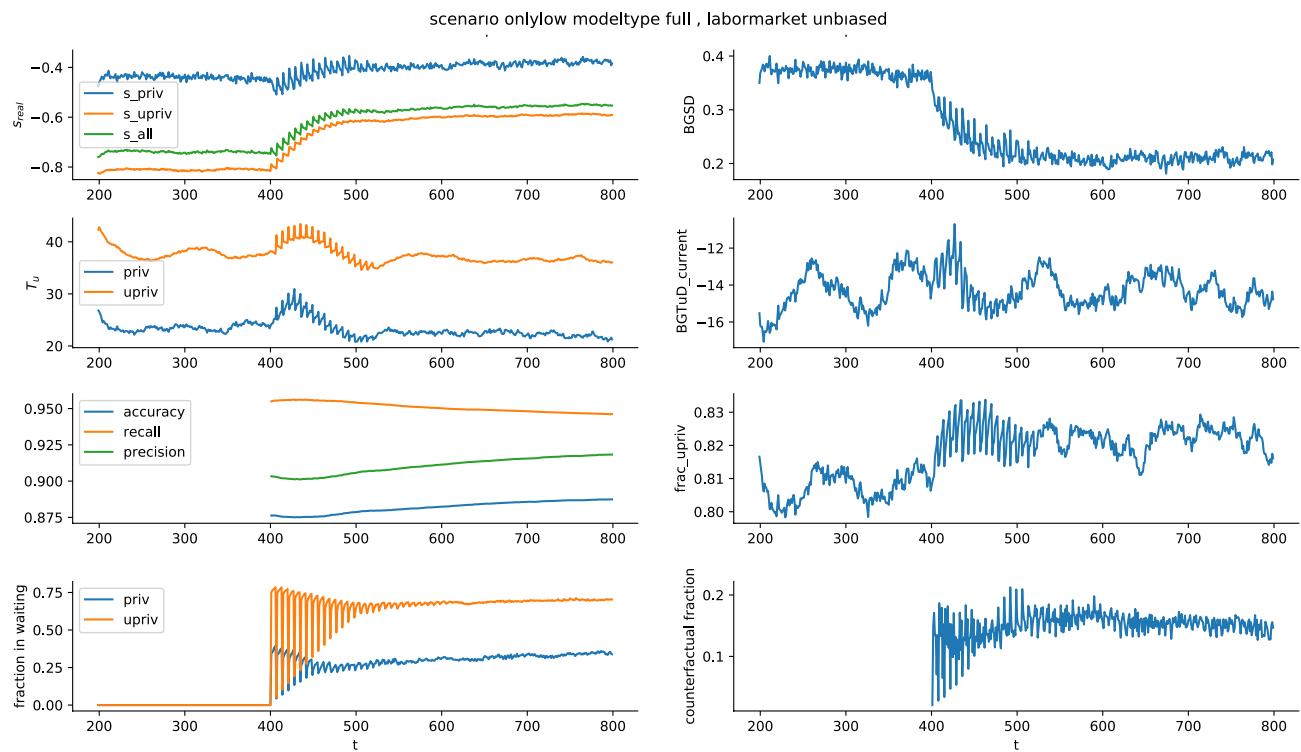


Figure 3. Time evolution of a single model run with the *onlylow* scenario and full (including protected attribute) prediction model. The First 200 timesteps are discarded, at timestep 400 the PES starts its intervention, which can be seen in several parameters (e.g. increase in s_{real} , decrease in $BGSD$). Some parameters/metrics are only available from the time the intervention starts.

from the background population, which does not compensate all the low-skilled individuals. After around 100 timesteps this reaches an equilibrium and the strong fluctuations vanish. Clearly visible is that a much higher fraction of the underprivileged group ends up in the waiting group compared to the privileged group.

Comparison of scenarios: influence on group skill. We now turn to a comparison of the different scenarios.

Figure 4 shows the time-evolution of $BGSD$, and Fig. 5 the average over the last 200 timesteps (a,b) and the difference between the average of the last and first 200 timesteps (c,d). The bars are split up by model type (full and base). In both figures, the left columns show the results for the runs with an unbiased labor market, and the right columns the results for the runs with a biased labor market. For the unbiased labor market, the PES with the base model has basically no effect on $BGSD$. With a biased labor market, the PES with base model *does* have an effect, but only for the intervention scenario in which only the high prospect group receives help. Interestingly, the $BGSD$ *decreases* in this scenario. This is a relatively counter-intuitive result. Therefore, we inspect it in more detail. The full model evolution plots are shown in the SI (Fig. C1–C2). The skills of the privileged group actually decrease. This can be explained the following way: the high prospect group receives help, which will affect the privileged group more. So there is a large number with very high skills, and therefore near 1 probability of finding a job in the first timestep after entering the pool of job seekers. Additionally, the labor market is biased towards the privileged group, therefore also individuals with intermediate skills have a relatively large chance of finding a job immediately. Thus, for the pool of job seekers (both active and waiting) on which we measure $BGSD$, $BGSD$ actually slightly decreases, as only the low-skilled ones from the privileged group remain in the pool. The result must also be connected with the fact that the individuals' low prospect group are put in the waiting group for $\Delta T_u = 5$ timesteps in the default model configuration, as when ΔT_u is set to zero, $BGSD$ does not decrease in the high-only scenario (not shown).

If the PES uses the full prediction model (which includes the protected attribute as predictor), $BGSD$ decreases for all scenarios, both in the unbiased and the biased labor market (Figs. 4c,d, 5c,d). The decrease in $BGSD$ is smallest for the *onlyhigh* scenario. The other three scenarios have a very similar larger decrease in $BGSD$, which is larger in the biased labor market. This clearly shows that the targeting does affect the influence on between group differences. Another interesting result is that none of the scenarios reaches a $BGSD$ of zero.

Comparison of scenarios: fairness of prediction model. We now turn to *counterfactual fraction* and *equal opportunity*, which measure different fairness goals than $BGSD$. In contrast to $BGSD$, they give insights into the fairness of the predictions model of the PES. The values for the end of the simulation are shown in Fig. 6. For the base model, *counterfactual fraction* is per definition zero, as the prediction model does not use the

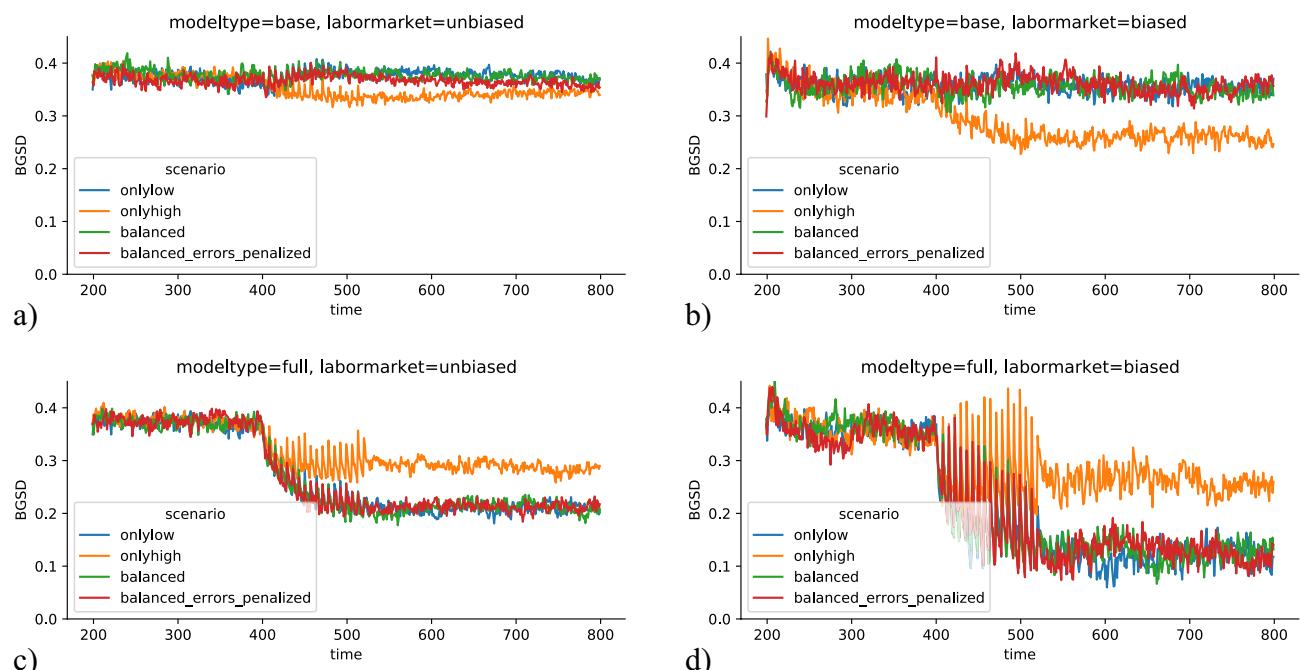


Figure 4. Time evolution of BGSD for all scenarios and all model combinations (full and base model, biased and unbiased prediction model). In an unbiased labor market and with the base prediction model, BGSD does not change significantly over time. Also, if the labor market is biased against the underprivileged group, BGSD does not change if the base prediction model is used, except if the help is targeted towards the high prospect group (scenario *onlyhigh*). If the full prediction model is used, BGSD decreases in all scenarios, independent of whether the labor market is biased or unbiased.

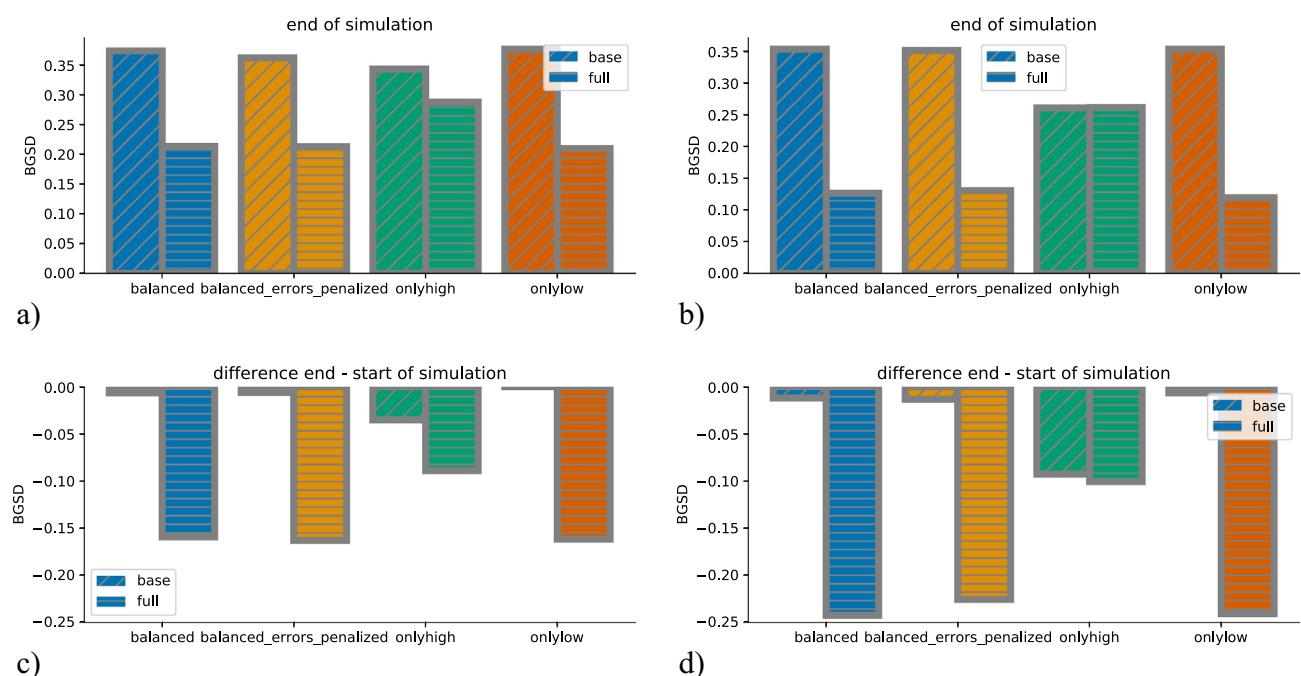


Figure 5. BGSD at the end of the simulations (a, b) and change of BGSD from start to end of simulations (c, d), for all intervention scenarios. a and c show the simulations for the unbiased, (b and d) the simulations for the biased labor market. Different colors indicate different intervention scenarios, and different hatching indicates the base (without protected attribute) and the full model (protected attribute).

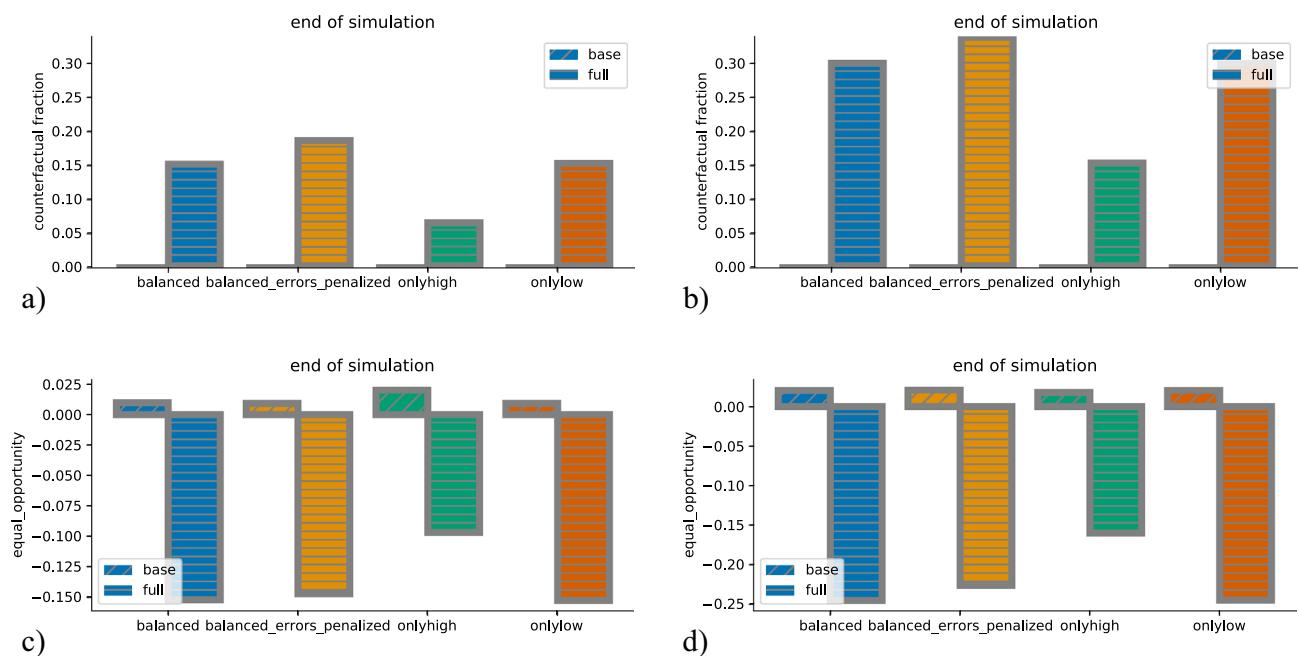


Figure 6. (a, b): Counterfactual fraction at the end of the simulations, for the unbiased (a) and biased (b) labor market. Different colors indicate different intervention scenarios, different hatching of the base (without protected attribute) and the full model (protected attribute). With the base prediction model, the *counterfactual fraction* is always zero. With the full prediction model, it is always positive, and depending on the intervention scenario. (c, d): Equal Opportunity ($TNR_{priv} - TNR_{upriv}$) at the end of the simulations, for the unbiased (c) and biased (d) labor market. Positive values indicate a positive bias in favor of the privileged group, and negative values a bias in favor of the unprivileged group.

protected attribute, and changing it, therefore, has no effect. For the full model, on the other hand, changing the protected attribute has an effect. In the unbiased labor market, roughly 10–20% of the individuals classified as low prospect are in fact high prospect and are only classified as low prospect because of their protected attribute. In the biased labor market, the percentage is roughly twice as high. The latter effect is in our eyes hard to foresee intuitively without explicit modeling. In comparison to the results with respect to the *BGSD* metric clearly shows that there is a non-negligible trade-off between the two fairness goals. The full prediction model is better suited for decreasing *BGSD* (which is one fairness goal) in 3 of the 4 scenarios, but it introduces a non-zero *counterfactual fraction* (whose avoidance is another fairness goal). The second effect the full prediction model has is that it changes equal opportunity from slightly positive values (better for privileged group) to negative values (worse for the privileged group—or more specifically, the true negative rate for the privileged group is smaller than for the underprivileged group).

Discussion and limitations

In this study, we have taken a simplistic view of the labor market. Our personal skill model is just as complex as necessary to capture the main setting (skills unevenly distributed across two groups). Further, we assume that success in the labor market is based solely on individual skills and that observations of the labor market (e.g., who finds a job) are thus a measure of skills. In reality, this view has been challenged fundamentally, as luck is just as important for success as individual skills and abilities (see⁴¹ and references therein). Additionally, personal factors such as sympathy can play a major role when selecting a candidate for an open position but will not be coded in the personal attributes potentially used by a PES (there is, however, the possibility that personal sympathy correlates with group-membership, thus further complicating things). Another important factor to consider is that both our models made the implicit assumption that there is, in principle, no shortage of jobs. If people have a high enough skill level, they will get a job (possibly). In our setting, there are enough jobs for the number of people, but not necessarily for their skill level. Real-world job-markets can, however, be limited on the side of open jobs, which would likely change the dynamics and effect of the PES's intervention.

While we included the option of having either a biased or unbiased labor market, the selection was fixed during the course of our simulations. In the real world, structural issues, such as biases in recruiters against or towards particular groups, may change over time. This raises the question of how representative—and, in the end, useful—historical recruitment data can be.

Also, in our abstract model-setting, the intervention scenarios we defined and applied in the study are only a subset of possible scenarios. In reality, even more different PES approaches would be conceivable, and this would need to be carefully reflected in the model setup.

Our intervention model (Eqs. (9) and (10)) is deterministic. In reality, the effect of the intervention (increase in skill) will also have a random component, which one could, for example, model by adding noise. We did not do this in our study in order to keep the model as simple and comprehensible as possible.

This study is, therefore, a proof of concept and not an analysis of a real-world system. To study an existing real-world system (such as the Austrian AMS system), one would need to (i) have access to the data the PES uses—or at least to aggregated statistics—and (ii) carefully model the dynamics of the respective labor market. Here, the trade-off between complexity and completeness would need to be addressed in greater detail.

Despite the given limitations, we believe that our findings could inform the design of real-world systems reflecting labor markets. For example, we have shown that there exists a trade-off between reducing the disparity between a privileged and an unprivileged group and misclassifying individuals. This fundamental trade-off between group fairness and individual fairness reflects an important aspect that system designers need to take into account. One potential solution could be to focus on group fairness but implement additional measures to validate and mitigate potential misclassifications of individuals.

Conclusion and future work

In this study, we investigated the long-term effects of data-driven intervention on the labor market in a simulated setting. Our results revealed an essential trade-off dilemma: the full model—in contrast to the base model—reduces BGSD, but at the same time classifies a number of individuals incorrectly as low-prospect solely because of their protected attribute, i.e., discriminates against them. Therefore, there is a trade-off between reducing the disparity between the two groups (reflected by a decrease in BGSD) and potentially treating individuals unfairly based on a protected attribute. Additionally, we found that active targeting of help (i.e., strategically distributing who receives what help) by the PES—compared to untargeted help—has little impact on inequality in the long-term, unless the help is targeted toward individuals with already high prospects, in which case inequality declines less. The purpose of this study was to show that in order to assess the ethical consequences of data-driven targeted support, e.g., for job-seekers, the investigation of long-term dynamics is crucial and requires careful quantitative modeling. This is not to say that other approaches (static quantitative approaches, philosophical/sociological approaches) are of less value, but that several perspectives are needed to give a complete picture. We have demonstrated this via a simple model for an employment market. Even in this relatively simple setting, it is not possible to answer questions on long-term fairness without explicit modeling.

A view on the long-term dynamics can be used for a more informed decision on whether to use a targeted support system. It can, however, also provide the basis for corrective actions that counteract unwanted long-term effects. Ideally, one would already consider long-term dynamics in the design phase. This is in line with⁴² who propose the implementation of ethics by design rules, particularly in respect to biases, values, and the effect of modern technological development on individuals, and more general initiatives for ethically aligned design⁴³.

With clearly defined long-term goals and constraints and an accurate model for the long-term dynamics, data-driven targeted support systems could from the beginning on be designed in a way that prevents—or at least minimizes the risk of—unfair outcomes over all relevant timescales.

Future work should focus on enhancing and adopting the model to better reflect real-world situations of labor market interventions by Public Employment Services, e.g., by investigating settings with more than two real skill features and one protected feature. Furthermore, even in the setting, we studied here, there are a number of additional fairness-related questions that are worthy of being addressed. For example: what is the long-term effect of targeted help on general employment? What is the long-term effect on employment in each group? Are there trade-offs between the—ethically problematic— inclusion of protected attributes in the targeting versus the global goal of high employment?

In this study, the effects of prescribed intervention scenarios were studied. A different approach would be to reverse the problem and use reinforcement learning to find strategies that the PES can use for achieving certain goals.

Finally, the same approach—careful quantitative dynamical modeling—may be applied to other similar problems of distributing public resources, for example, in the context of education or public funding.

Data availability

The software for this study was written in Python and is published in the first author's personal GitHub repository (<https://github.com/sipposip/jobservice-ads-lterm-impact>) and as FOSS via Zenodo (<https://zenodo.org/record/6962331>). It allows full reproduction of the results of this study as well as further experimentation with the model parameters.

Received: 22 August 2022; Accepted: 25 January 2023

Published online: 31 January 2023

References

- Berryhill, J., Heang, K. K., Clogher, R. & McBride, K. Hello, World: Artificial intelligence and its use in the public sector. *OECD Publishing*<https://doi.org/10.1787/726fd39d-en> (2019).
- Wong, K. L. X. & Dobson, A. S. We're just data: Exploring China's social credit system in relation to digital platform ratings cultures in westernised democracies. *Global Media China* **4**, 220–232. <https://doi.org/10.1177/2059436419856090> (2019).
- Datta, A., Tschantz, M. C. & Datta, A. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination. arXiv preprint [arXiv:1408.6491](https://arxiv.org/abs/1408.6491) (2014).
- Friedman, B. & Nissenbaum, H. Bias in computer systems. *ACM Trans. Inf. Syst. (TOIS)* **14**, 330–347 (1996).
- Eubanks, V. *Automating inequality: How high-tech tools profile, police, and punish the poor* (St. Martin's Press, 2018).
- Commission, E. Proposal for a regulation of the European parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain Union legislative acts. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206> (2021).
- Lippens, L., Baert, S., Ghekiere, A., Verhaeghe, P.-P. & Derous, E. Is labour market discrimination against ethnic minorities better explained by taste or statistics? A systematic review of the empirical evidence. *IZA Discuss. Pap.* **48**, 4243–4276 (2020).

8. Petrongolo, B. The gender gap in employment and wages. *Nat. Hum. Behav.* **3**, 316–318 (2019).
9. Holl, J., Kernbeiß, G. & Wagner-Pinter, M. Das AMS-Arbeitsmarkt-chancen-Modell.
10. Lopez, P. Reinforcing intersectional inequality via the AMS algorithm in Austria (2019).
11. Allhutter, D., Cech, F., Fischer, F., Grill, G. & Mager, A. Algorithmic profiling of job seekers in Austria: How Austerity politics are made effective. *Front. Big Data* **3**, 5. <https://doi.org/10.3389/fdata.2020.00005> (2020).
12. Allhutter, D., Mager, A., Cech, F., Fischer, F. & Grill, G. DER AMS-ALGORITHMUS Eine Soziotechnische Analyse des Arbeitsmarktchancen-Assistenz-Systems (AMAS). ITA-Projektbericht (2020).
13. D'Amour, A. et al. Fairness is not static: Deeper understanding of long term fairness via simulation studies. In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, 525–534, (Association for Computing Machinery, New York, NY, USA, 2020). 10.1145/3351095.3372878.
14. Bagdasaryan, E., Poursaeed, O. & Shmatikov, V. Differential privacy has disparate impact on model accuracy. In: Wallach, H. et al. (eds.) *Advances in Neural Information Processing Systems*, vol. 32 (Curran Associates, Inc., 2019).
15. Xenidis, R. & Senden, L. Eu non-discrimination law in the era of artificial intelligence: Mapping the challenges of algorithmic discrimination. *General Principles of EU law and the EU Digital Order* (Kluwer Law International, 2020) 151–182 (2019).
16. Commission, E. White paper on artificial intelligence—a European approach to excellence and trust. *Com* (2020) 65 Final (2020).
17. Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudik, M. & Wallach, H. Improving fairness in machine learning systems: What do industry practitioners need? In: *Proceedings of the 2019 CHI conference on human factors in computing systems* 1–16 (2019).
18. Madaio, M. A., Stark, L., Wortman Vaughan, J. & Wallach, H. Co-designing checklists to understand organizational challenges and opportunities around fairness in ai. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* 1–14 (2020).
19. Arrow, K. The Theory of Discrimination. Working Paper 403, Princeton University, Department of Economics, Industrial Relations Section. (1971).
20. European Union, Agency for Fundamental Rights. #BigData: Discrimination in data-supported decision making. 10.1163/2210-7975_HRD-9992-20180020 (2018).
21. European Union, Agency for fundamental rights. Data quality and artificial intelligence – Mitigating bias and error to protect fundamental rights (Publications Office of the European Union, 2019), 978-92-9474-606-1 edn.
22. Union, European. *Agency for Fundamental Rights* (Report (Publications Office of the European Union, Getting the Future Right Artificial Intelligence and Fundamental Rights, 2020).
23. Wachter, S., Mittelstadt, B. & Russell, C. Why fairness cannot be automated: Bridging the gap between EU non-discrimination law and AI. *SSRN Electron. J.* <https://doi.org/10.2139/ssrn.3547922> (2020).
24. Bellamy, R. K. E. et al. AI fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *CoRR* abs/1810.01943 (2018).
25. Verma, S. & Rubin, J. Fairness definitions explained. In: *2018 IEEE/acm international workshop on software fairness (fairware)*, 1–7 (IEEE, 2018).
26. Srivastava, M., Heidari, H. & Krause, A. Mathematical notions vs. human perception of fairness: A descriptive approach to fairness for machine learning. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* 2459–2468 (2019).
27. Liu, L. T., Dean, S., Rolf, E., Simchowitz, M. & Hardt, M. Delayed impact of fair machine learning. In: *International Conference on Machine Learning* 3150–3158 (PMLR, 2018).
28. Mouzannar, H., Ohannessian, M. I. & Srebro, N. From fair decision making to social equality. In: *Proceedings of the Conference on Fairness, Accountability, and Transparency* FAT* '19, 359–368 (Association for Computing Machinery, 2019) 10.1145/3287560.3287599 .
29. Kannan, S., Roth, A. & Ziani, J. Downstream effects of affirmative action. In: *Proceedings of the Conference on Fairness, Accountability, and Transparency* FAT* '19, 240–248 (Association for Computing Machinery, 2019) 10.1145/3287560.3287578.
30. Binns, R. On the apparent conflict between individual and group fairness. In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* 514–524 (2020).
31. Goetz, S. J. Labor market theory and models. In: Fischer, M. M. & Nijkamp, P. (eds.) *Handbook of Regional Science* 35–57 (Springer, Berlin, 2014) 10.1007/978-3-642-23430-9_6.
32. Neugart, M. & Richiardi, M. *Agent-Based Models of the Labor Market*, vol. 1 (Oxford University Press, 2018).
33. Chaturvedi, A., Mehta, S., Dolk, D. & Ayer, R. Agent-based simulation for computational experimentation: Developing an artificial labor market. *Eur. J. Oper. Res.* **166**, 694–716. <https://doi.org/10.1016/j.ejor.2004.03.040> (2005).
34. Cain, G. G. Chapter 13 the economic analysis of labor market discrimination: A survey. vol. 1 of *Handbook of Labor Economics*, 693–785 (Elsevier, 1986) 10.1016/S1573-4463(86)01016-7.
35. Lang, K. & Lehmann, J.-Y.K. Racial discrimination in the labor market: Theory and empirics. *J. Econ. Lit.* **50**, 959–1006. <https://doi.org/10.1257/jel.50.4.959> (2012).
36. Ehrenberg, R. G., Smith, R. S. & Hallock, K. F. *Modern labor economics: Theory and public policy* (Routledge, 2021).
37. Hu, L. & Chen, Y. A short-term intervention for long-term fairness in the labor market. In: *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18* 1389–1398 (ACM Press, 2018) 10.1145/3178876.3186044.
38. Cohen, L., Lipton, Z. C. & Mansour, Y. Efficient candidate screening under multiple tests and implications for fairness. *CoRR* abs/1905.11361 (2019).
39. Biondo, A. P. A. E. & Rapisarda, A. Talent vs luck: The role of randomness in success and failure. *Adv. Complex Syst.* **21**, 1850014. <https://doi.org/10.1142/S0219525918500145> (2018).
40. Hardt, M., Price, E. & Srebro, N. Equality of opportunity in supervised learning. *Advances in neural information processing systems* **29** (2016).
41. Blattner, L. & Nelson, S. How costly is noise? Data and disparities in consumer credit. *arXiv:2105.07554* [cs, econ, q-fin] (2021).
42. Mulvenna, M., Boger, J. & Bond, R. Ethical by design: A manifesto. In: *Proceedings of the European Conference on Cognitive Ergonomics* 51–54 (2017).
43. on Ethics of Autonomous, T. I. G. I. & Systems, I. Ethically aligned design: A vision for prioritizing human well-being with autonomous and intelligent systems (2019).

Acknowledgements

This work was supported by the “DDAI” COMET Module within the COMET – Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry (BMK and BMDW), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG.

Disclaimer

We have used a color scale for our graphics that at the first glance might seem suboptimal, as the colors are seemingly not very easy to distinguish. We have, on purpose, opted for that color scale, as it is, to our knowledge, the best color scale for that number of colors that still can also be interpreted by individuals with red-green color vision deficiency.

Author contributions

S.S. conceived the idea for the study and developed the model. All authors contributed to the analysis, the interpretation of the results and the drafting of the manuscript.

Competing interest

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-023-28874-9>.

Correspondence and requests for materials should be addressed to S.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023