School of Informatics

Informatics Project Proposal
Recommender Systems: looking further into the future

B225254
April 2023

Abstract

Recommender systems are widely used to assist people in finding relevant information, products, and services. However, the standard approach of predicting just the next immediate action may not fully capture the complexity of users behaviors. The project aims to investigate whether predicting the sequence of users' future actions with a time-window predictor is a more effective learning task for recommendation systems than the more traditional way. We will develop a new way of capturing users behaviors and evaluate how well it works in making personalized recommendations. This approach will potentially increase the system's overall performance and decrease the need for model retraining and deployment.

Date: Wednesday 19th April, 2023

Tutor: Aurora Constantin
Supervisor: David Wardrope, Michael la Grange, Timos Korres

1 Motivation

Recommender systems have become an essential component of e-commerce platforms, entertainment apps, and other services in the recent years [1, 2, 3]. These systems offer personalized suggestions to users based on their preferences, historical interactions, and other behavioral data [3]. These systems have the potential to enhance customers satisfaction, increase subscriptions, and contribute to the growth of companies in various contexts, such as online retail (e.g. Amazon), and streaming services (e.g. Netflix) [3]. However, most recommendation systems focus on predicting just the user's next action, by taking into account the sequential nature of past behaviors, without considering their sequential future behaviors [1, 2, 4, 5, 6, 7].

When modelling the sequential user behavior, more accurate and timely recommendations can be made, especially for e-commerce shops where users purchase items in a sequence [8, 9, 5]. Predicting an ordered sequence of users' future purchases is an important goal in the field of sequential recommender systems, as it has the potential to significantly improve the recommendations made to users. For example, if a user purchases a phone, the next item that they might purchase is a charger, and after that a case. Many approaches have been explored in the literature in the context of sequential recommender systems, from using simple Markov Chain [10] to more advanced Transformer-based models (more in Section 2)[9]. Furthermore, having a window-based prediction to capture user's future behaviors can have an impact on industrial companies (such as Amazon and Netflix) [11]. More specifically, predicting the sequence of users' future purchases within a given time window, can allow effective offline recommendations at inference time, which will improve the system's overall performance, and reduce the frequency of model retraining & deployment [11]. This will potentially lead to a significant cost and resource reduction. Lastly, the research community could benefit from advances in this this study as it will help them understand more complex user behaviors and preferences over time.

In light of this, the main goal of this research is to investigate whether predicting the ordered sequence of user's future actions (purchases or movies to watch) is a better learning task for recommender systems than the standard approaches that just predict the next action. We aim to to build on top of the existing literature on sequential recommendations and develop a new novel way of capturing users' behaviors.

The project proposal is structured as follows: The rest of this section will discuss the research scope and objectives. Thereafter, Section 2 will demonstrate background work in the area. Sections 3 and 4 will provide the methodology and evaluation respectively. Moreover, Section 5 will outline the expected outcomes of the project. Lastly, Section 6 will give an overview of research plan, milestones, and deliverables.

1.1 Problem Statement

The current standard approach for recommender systems is to predict the users' next most likely behavior based on their past purchases or actions [1, 2, 4, 5, 6, 7, 8]. However, this way of learning user's behaviors, may not accurately capture the changing in their preferences over time [12]. Additionally, this approach might require more computational resources as the models will need frequent retraining and deployment [11]. Pancha et al. in their study [11] tried to address this problem by incorporating a new way of predicting user's future long-term engagement. Nonetheless, their future predictions are made by selecting random examples without considering relative time to the target prediction (they predict an unordered multiset of future behaviors) [11]. Furthermore, the authors use a fixed 28-day window to train the model which might limit the idea of capturing short-term preferences [11]. To tackle this drawback, we examine whether predicting the ordered sequence of a user's future actions within a given time window is a better learning tasks for recommender systems. More specifically, we will try to predict items a user will potentially interact within a time window, such as the next 5 minutes, 1 day, or 1 week. We will try to solve this by evaluating the effectiveness of this alternative approach, and identifying the strengths and weaknesses over the other existing methods [6, 11].

1.2 Research Hypothesis and Objectives

This study will look into whether predicting a user's future actions within a specific time frame can lead to more accurate personalized recommendations compared to standard methods like [2, 6, 7, 12]. Short-term changes in a user's behavior, such as those that occur within a day or a week, will be captured to increase the robustness of the model. The study aims to:

1. Develop a baseline sequential recommender system for next-item predictions.
2. Extend the system to predict the sequence of a user's future purchases.
3. Evaluate the effectiveness of using a time window predictor for making recommendations.
4. Determine the optimal time window[1] for predicting future behaviors.
5. Compare the proposed approach with state-of-the-art methods for recommendation methods.

In light of this, the proposed approach has the potential to increase recommender systems' reliability and accuracy by taking into account users' evolving behavior, leading to a more personalized recommendations. This approach could increase customers satisfaction and loyalty, as it will be more likely to find products they are interested in [3]. Lastly, this could have a substantial impact on e-commerce and other online services [3].

1.3 Timeliness and Novelty

While various studies [1, 2, 4, 5, 6, 7, 8] have been employing sequential recommender systems to capture users' behaviors and improve recommendations, there is only one research (Pinnerformer [11]) that has sought to employ a time window predictor to predict future item recommendations. Even though the authors in [11] demonstrated successfully a system to capture next 28-day items, there is more to explore to capture users' short-term preferences. The study will extend recent work in sequential recommender systems to further understand users' preferences and make more robust recommendations. The research will build on Kang et al. [6] and Pancha et al. [11] works to further improve capturing short- and long-term preferences of the users. Further, this study comes at a time when the researching field in personalization systems is emerging and new e-services around the world start to get attracted by their capabilities [3].

1.4 Significance

If successful, this study will demonstrate a novel way of addressing the limitations of the widely used next-item prediction approaches [12]. It will potentially shape a new understanding of

[1] Our initial hypothesis sets different time windows for different datasets.

users' preferences and behavior changes over time. Moreover, this new approach could improve the robustness, efficiency, and accuracy of the recommendation systems, as it aims to capture both short- and long-term behaviors of the customers and limit the need of retraining and deployment. Furthermore, the study will have a significant impact on current state of knowledge in the field of recommender systems by exploring a new way of learning and expanding recent studies, such as Kang et al. [6] and Pancha et al. [11].

1.5 Feasibility

The study needs careful planning as the the time allocated for it is limited. In Section 6, a detailed plan is demonstrated. The development of the time-window recommender system should take a big portion of the time from the project as it is the most challenging part. Nonetheless, utilizing a baseline will be straightforward because a system from the literature will be used. GitHub repositories with current state-of-the-art systems exist, such as SASRec[2], which could be utilized as baselines. Later on, to extend the current studies and implement the time-window predictor, similar recommender models can be extended directly from off the shelf tools and frameworks to save more time. Different datasets are already public available to the research community, therefore it will be straight forward using and pre-processing those. Upon the development of the model, GPUs via the MLP clusters are provided, which will help speed up the training process. If problems with the MLP cluster raise, other cloud machines should be considered. Evaluation of the system after training requires careful consideration due to limited existing studies on future item prediction.

1.6 Beneficiaries

As aforementioned, the proposed project offers many advantages. The findings of this study will shed light on the way the researching field thinks about humans' behaviors when it comes to personalization. Predicting the sequence of users' future actions compared to next item predictions can potentially open new ways of how recommendation system should work. This study will be leveraged not only by researches, but also companies in the industry such as e-commerce, advertising, and other services [3]. Additionally, building such a good recommendation system, will increase the satisfaction of customers around the world. Upon the successfully completion of the study, to ensure that other researches can benefit from this, an academic publication will be released as a stretched goal of the study. In addition to that, the source code will be made publicly available in order for others to join and extend the findings.

2 Background and Related Work

Recommendation systems have been explored broadly in the literature as they have become important due to the demand for personalized services [3, 12]. The ultimate goal of these systems is to give relevant recommendations based on their personalized behaviors and preferences [3]. Traditional recommendation systems use collaborative filtering or content-based approaches to make recommendations to users [9, 12, 13]. According to the theory behind collaborative filtering, users with similar preferences will continue to have those same preferences in the future [14, 13]. On the other hand, content-based filtering suggests items that are comparable to those that the user has previously liked [13, 15]. These systems, tend to use explicit (e.g. purchases) or

[2] SASRec repository https://github.com/kang205/SASRec/ [6]

implicit (e.g. clicks) interactions of the users, and they assume that all interactions are equally important for learning their preferences [12]. However, in real-world situations, a user's next action depends by both their long-term preference, and their current intent that is influenced by recent interactions [12, 9]. To tackle this problem, recent academic research has focused on sequential recommender systems, which enhance personalizations by incorporating temporal and sequential information [3, 2, 11, 1, 12]. Sequential recommendation systems can capture complex patterns and understand users better by handling long-term and short-term behavior changes, ensuring more robust recommendations [12].

2.1 Traditional Approaches

Traditional sequential recommendation systems, apply collaborative filtering (CF) and Markov Chains (MC) to make recommendations. As aforementioned, CF makes the assumption that similar users behaviors will do the same next time based only on statistical methods [8, 14, 16]. A simple CF method used across the literature as a baseline is Bayesian Personalized Ranking (BPR), which uses matrix factorization to learn personalized rankings based on their interactions [6, 2, 7, 17]. CF methods usually tend to under-perform due to the sparse data and cold-start problems (new users with limited data) [3, 14]. One of the methods introduced to capture sequential behaviors are Markov Chain methods [3, 8, 10]. MC-based models are able to model short-term interests in sequential patterns using context information, such as the Fossil model [18, 8], something that makes the MC models effective at predicting the next sequence item [8]. However, MC only takes into account the correlations between elements that are close to each other and cannot capture the long-term users' preferences and changes over time [8, 19].

2.2 Deep Learning Approaches

In recent years, deep learning methods have emerged as promising approaches to address traditional methods limitations and enhance personalization [1, 2, 3, 4, 6, 7, 11, 12]. Multi-layer Perceptions (MLPs) have been widely used [20]. However, using MLPs is not recommended anymore because selecting appropriate hyper-parameters, such as network depth remains a challenge [8]. Convolutional Neural Networks (CNNs) have been utilized in the literature, where they have been applied to extract features from various data sources, such as text, images, and audios to improve recommendations [8, 20, 21, 22]. One example of using CNNs in sequential recommender systems is the Caser model introduced by Jiaxi Tang et al., which used the network to generate low-dimensional embeddings of user-item interactions [23]. Nonetheless, researches found that insufficient data limits their ability to capture meaningful information [8].

Recurrent Neural Networks (RNNs) showed to be really effective in recommendation tasks as they are capable of capturing dynamic time series data [8, 24, 25, 26]. To take into account long-term preferences and weight them properly, other RNN variants such as LSTMs [26, 27], and Gated Recurrent Units (GRU) [28] use gating mechanisms [8]. Hidasi et al. [7] proposed GRU4Rec, which uses GRU to effectively model user-items interaction sequences. The researchers have then proposed GRU4Rec+ [29] system that integrates new loss functions named TOP1-max and BPR-max, as well an enhanced sampling strategy to boost the performance [9]. More specifically, the authors combined successfully the Top1-max loss, which maximizes the probability of predicting the correct item, and the BPR-max, which tries to maximize the ranking of the correct item [7, 29]. Although RNN-based models have demonstrated their effectiveness in recommendation tasks, they still carry significant limitations [8, 9].

Attention mechanisms have proven to be effective in different domains such as machine translation [30] and image captioning [31]. They key idea behind them is to identify relevant parts of the input that contribute to the output [2]. In light of this, attention mechanisms have also been introduced in sequential recommender systems to identify relevant items based on the user's historical behaviors [2, 3, 8, 11]. Liu et al. [32] introduced a novel a short-term attention/memory priority model (STAMP) that uses vanilla attention mechanism to determine the attention scores between the last item and next items in the sequence [32]. Another state-of-the-art method is SASRec, introduced by Kang et al.[6]. The authors employed a self-attention mechanism to balance the short-term intentions and long-term preferences of the users [6, 12]. SASRec is used as an advanced baseline in most recent studies in the field [2, 8, 11].

An advanced sequential recommendation model known as BERT4Rec [33] has incorporated the transformer architecture to enhance its performance. It trains a bidirectional model for modeling sequential data by utilizing the Cloze task [33]. Cloze task masks out items in a sequence and train the model to predict the masked items to improve the performance [33]. Moreover, TiSASRec [2] enhances SASRec by incorporating time intervals between items that appear in the sequence [2, 12]. The model generates a relation matrix between items based on the time intervals of each pair of items in a user's historical sequences [2, 12]. However, TiSASRec is limited in its ability to understand differences between timestamps in various contexts and ignores the consistent behaviors in similar timestamps [2, 9]. Zhang et al. [9] to tackle this, they proposed TAT4SRec. TAT4SRec incorporates an encoder-decoder model which split timestamps and interacted items to improve temporal information. TAT4SRec also employs two embedding modules, including a window-based function to preserve continuous dependencies in similar timestamps [9]. To tackle the computational limitation of RNN-based systems, Sun et al.[34] in their study employed a transformer model to boost training time and capture the connection between items without regards to the distance [12, 34]. The authors in [8] outperformed other state-of-the-art baselines by creating a time-aware long- and short-term (TLSAN) model [8]. The model uses personalized time-aggregation to capture users' personalized long-term habits, and uses attentions to capture the short-term behaviors [8].

Finally, Pancha et al. proposed Pinnerformer [11] to address the challenge of deploying large models in production and managing mutable data [11]. Pinnerformer builds on top of Pinnersage [35] and introduces a novel window-based prediction approach [11]. Their goal is to train the model learn to predict user's positive future actions over 28 day window [11]. To achieve that, the authors applied a novel novel sampled softmax with logQ loss function, which computes a weighted average loss for each user and positive embedding pair. The correction term accounts for the probability of a negative example appearing in the batch [11]. The model used, utilizes a transformer architecture to generate a sequence of user representations [11]. They use the final and intermediate user representations to predict a random positive action within a time window during inference by generating new embeddings for users who have engaged with anything the last day. They merge them with the previous day's embeddings, and then store them as key-values to make recommendations [11]. Authors call it "dense all action prediction" [11].

3 Programme and Methodology

As mentioned previously, this study will focus on developing a novel sequential recommender system with a time window predictor to enhance the accuracy and efficiency of the system. In order to work on our high-level objective, we will break it into a number of work packages, as illustrated in the following aspects. The project will be managed using agile methodology, with weekly supervision meetings. Section 6 gives a detailed project plan outlining the milestones and deliverables of the project, and a Gantt chart to monitor progress through the weeks.

Data Collection & Preparation: In order to build a recommender system, data collection and preparation should be carried out. We can collect potential datasets, such as Amazon [3] and MovieLens [36] from online available sources[3] and use them to build the system. However, after collecting potential datasets, exploratory data analysis (EDA) should be carried out to determine if the potential dataset can actually be used to build a multi-day window predictor. Our system requires relatively long sequences of user-items interactions in order to ensure robustness in predicting multiple items in the future. For example, Kang et al. in the SASRec study, kept only sequences that have more than 5 interactions [6]. Because this might be a problem, some data augmentation techniques like noise injection, item masking, and synonym replacement might be worth looking into to enhance the system [37]. Apart from enhancing our data, different splitting strategies should be explored [38].

Baseline System Development: After completing data collection & preparation, the baseline model will be implemented. We will use the standard approach of predicting the next item for the baseline system as in [6, 2]. SASRec [6] state-of-the-art model will be potentially utilized straight out off the shelf for this work package. Upon the successful implementation the baseline should be evaluated using metrics such as Recall, and Normalized Discounted Cumulative Gain (NDCG) [12]. Lastly, different data strategies from the previous work package could be explored.

Proposed System Development: The next step of the project is to extend the baseline system and incorporate a time window predictor. The first step towards developing such system is to define how the model will be used. More specifically, with regards to the dataset, we should determine the optimal time window to predict items. Some categories on the Amazon dataset might require different time window as the behaviors of the customers change. For instance, Amazon Beauty category might draw users' interest weekly rather than daily, thus a week time window might be more suitable. After that, we should utilise a proper technique to help our trained system to be accurate in making future predictions. An example of a technique to use is the Cloze task. Incorporating the Cloze task from BERT4Rec [33] to mask items may potentially help our proposed model. Instead of masking randomly items as they do in the paper, it would be interesting looking into masking items relative to the time target we are planning to have in our time window model. More specifically, the model would mask items based on their temporal proximity to the target time window, something that would ensure that the model learns to predict items that are more likely to be relevant to the user at the target time, thereby improving the system's accuracy in making future predictions. Furthermore, the recent sampled softmax logQ loss function used in the Pinnerformer paper [11] could be employed, as it demonstrated successfully good performance in similar tasks of predicting items in the next 28 days. Other techniques worth exploring could be scheduled sampling [39] or recency-based sampling for sequences [40] to help model handle uncertainty and variability in users' behaviors, and add a form of regularization. Different embedding strategies will be utilized to ensure effective training. Something similar to Pinnerformer [11] could be done. Apart from positional user/items embeddings and raw absolute timestamps, time of the most recent action, and the intervals between actions should be encoded and fed to the model [11]. Encoding methods such word2vec, item2vec, and time2vec could be utilized [41, 42, 43]. At inference time, the system could also incorporate Pinnerformer's way to merge last user embeddings with previous day's embeddings [11].

Performance Evaluation and Analysis: After completing the development work package, we will evaluate the performance of both models using standard metrics such as Recall@k,

[3] Amazon Dataset, MovieLens Dataset

NDCG, and Mean Reciprocal Rank (MRR) [12]. At inference time, each model will be asked to predict a list of k items at a specific future time to make recommendations. These recommendations, will then be compared to the truth set of items that user actually interacted with using the aforementioned metrics. Evaluation will give us a better understand in order to further optimize the proposed system.

Limitations: While the study aims to develop an accurate and robust recommendation system, it will be trained on specific datasets. This can limit the work as other datasets in the area may not fit to the proposed model, leading to poor performance. Also, some datasets might have other optimal time window for predictions. This should be considered for future work and reuseability as not all the scenarios will be explored in this study.

3.1 Risk Assessment

The following table presents the risks of the study and how can they be mitigated.

Description	**Impact**	**Likelihood**	**Mitigation**
Data limitations (poor quality and/or insufficient data)	High	Medium	Data exploration and analysis to identify issues. Apply data augmentation. New data sources.
Time constraints with study's scope	High	Low	Break down project into smaller tasks with clear deadlines. Prioritize tasks. Rethink project plan.
Low model performance	High	High	Retrain models regularly, use sampling techniques, explore new model architectures, generate new embeddings.
Computational resources limitations	Medium	Medium	Optimization of model architecture and hyperparameters. Use cloud-based computing resources.

Table 1: Risk Assessment

3.2 Ethics

Since we will be using publicly available datasets used in the literature [1, 2, 3, 4, 6, 7, 11, 12], there are not any significant ethical concerns. However, unintentional bias towards users with different gender, ethnicity, and/or age needs to be considered during training to avoid cases like recommending constantly romantic comedies to women and action movies to men. In such cases, new dataset, or preprocessing to eliminate some bias should be considered.

4 Evaluation

To evaluate the performance of the proposed sequential recommender system, we plan to use and test two datasets. We will evaluate our system on the Amazon [3] and MovieLens [36] datasets as aforementioned. The raw features we are planning to use from these datasets include user IDs, item IDs, and timestamps. For evaluating the performance of the proposed sequential recommender system, we plan to use widely adopted metrics that can be seen in the field [2, 6, 11, 12, 44]. The most seen metrics are: Recall@k, MRR, NDCG, and Hit Ratio (HR). Inspired by Pinnerformer, we can use recall@k_t where k represents the number of candidates considered at the timestamp t in the future time window, to evaluate the performance [11].

5 Expected Outcomes

By doing this study we aim to shed light on whether using a time window for future predictions by a recommender system is a better learning task compared to existing approaches that predict the next single item. After Pancha et al. [11] extended their recommender system to make future predictions successfully, we hope that we can develop a system which can capture long- and short-term users' preferences effectively, and make more accurate and robust recommendations at inference time that are also less susceptible to short-term variations. Furthermore, the proposed system will be resource-friendly as less frequent retraining & deployment will be required. Upon a successful completion, this research could make a significant contribution to the field of recommender systems, in both academic and industrial settings.

6 Research Plan, Milestones and Deliverables

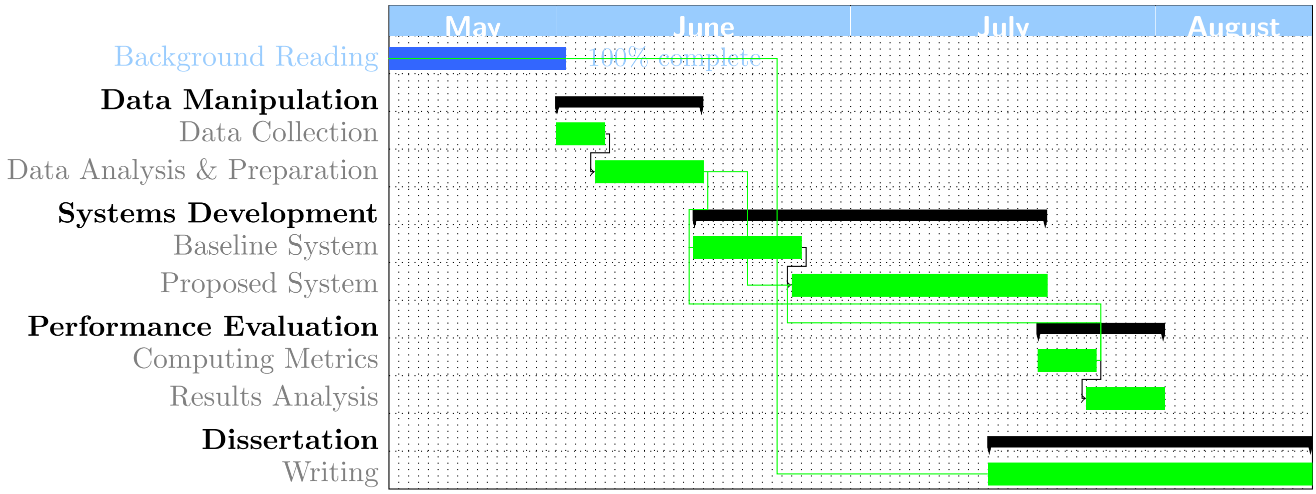

Figure 1: Gantt Chart of the activities defined for this project.

Milestone	Week	Description
M_1	2	Data Manipulation completed
M_2	4	Development of baseline system completed
M_3	7	Development of proposed system completed
M_4	9	Evaluation completed
M_5	12	Submission of dissertation

Table 2: Milestones defined in this project.

Deliverable	Week	Description
D_1	4	Baseline System
D_2	7	Proposed System
D_3	9	Evaluation report
D_4	12	Dissertation

Table 3: List of deliverables defined in this project.

References

[1] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. Sequential recommender systems: challenges, progress and prospects. *arXiv preprint arXiv:2001.04830*, 2019.

[2] Jiacheng Li, Yujie Wang, and Julian McAuley. Time interval aware self-attention for sequential recommendation. In *Proceedings of the 13th international conference on web search and data mining*, pages 322–330, 2020.

[3] Julian McAuley. *Personalized Machine Learning*. Cambridge University Press, 2022.

[4] Wendi Ji, Keqiang Wang, Xiaoling Wang, Tingwei Chen, and Alexandra Cristea. Sequential recommender via time-aware attentive memory network. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 565–574, 2020.

[5] Yongfu Zha, Yongjian Zhang, Zhixin Liu, and Yumin Dong. Self-attention based time-rating-aware context recommender system. *Computational Intelligence & Neuroscience*, 2022.

[6] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE, 2018.

[7] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

[8] Jianqing Zhang, Dongjing Wang, and Dongjin Yu. Tlsan: Time-aware long-and short-term attention network for next-item recommendation. *Neurocomputing*, 441:179–191, 2021.

[9] Yihu Zhang, Bo Yang, Haodong Liu, and Dongsheng Li. A time-aware self-attention based neural network model for sequential recommendation. *Applied Soft Computing*, 133:109894, 2023.

[10] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, et al. The youtube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 293–296, 2010.

[11] Nikil Pancha, Andrew Zhai, Jure Leskovec, and Charles Rosenberg. Pinnerformer: Sequence modeling for user representation at pinterest. *arXiv preprint arXiv:2205.04507*, 2022.

[12] Hui Fang, Danning Zhang, Yiheng Shu, and Guibing Guo. Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. *ACM Transactions on Information Systems (TOIS)*, 39(1):1–42, 2020.

[13] Poonam B Thorat, Rajeshwari M Goudar, and Sunita Barve. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4):31–36, 2015.

[14] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009, 2009.

[15] Robin Van Meteren and Maarten Van Someren. Using content-based filtering for recommendation. In *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, volume 30, pages 47–56. Barcelona, 2000.

[16] Lyle Ungar and Dean P Foster. A formal statistical approach to collaborative filtering. *CONALD'98*, 1998.

[17] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.

[18] Ruining He and Julian McAuley. Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 191–200. IEEE, 2016.

[19] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820, 2010.

[20] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. *Advances in neural information processing systems*, 26, 2013.

[21] Ruining He and Julian McAuley. Vbpr: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[22] Pei-Yi Hao, Weng-Hang Cheang, and Jung-Hsien Chiang. Real-time event embedding for poi recommendation. *Neurocomputing*, 349:1–11, 2019.

[23] Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 565–573, 2018.

[24] Maarten De Rijke. *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. Association for Computing Machinery, 2017.

[25] Yaqing Wang, Caili Guo, Yunfei Chu, Jenq-Neng Hwang, and Chunyan Feng. A cross-domain hierarchical recurrent model for personalized session-based recommendations. *Neurocomputing*, 380:271–284, 2020.

[26] Yuwen Zhou, Changqin Huang, Qintai Hu, Jia Zhu, and Yong Tang. Personalized learning full-path recommendation model based on lstm neural networks. *Information sciences*, 444:135–152, 2018.

[27] Chuanchuan Zhao, Jinguo You, Xinxian Wen, and Xiaowu Li. Deep bi-lstm networks for sequential recommendation. *Entropy*, 22(8):870, 2020.

[28] Anjing Luo, Pengpeng Zhao, Yanchi Liu, Jiajie Xu, Zhixu Li, Lei Zhao, Victor S Sheng, and Zhiming Cui. Adaptive attention-aware gated recurrent unit for sequential recommendation. In *Database Systems for Advanced Applications: 24th International Conference, DASFAA 2019, Chiang Mai, Thailand, April 22–25, 2019, Proceedings, Part II 24*, pages 317–332. Springer, 2019.

[29] 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.

[30] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[31] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[32] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 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*, pages 1831–1839, 2018.

[33] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.

[34] Shiming Sun, Yuanhe Tang, Zemei Dai, and Fu Zhou. Self-attention network for session-based recommendation with streaming data input. *IEEE Access*, 7:110499–110509, 2019.

[35] Aditya Pal, Chantat Eksombatchai, Yitong Zhou, Bo Zhao, Charles Rosenberg, and Jure Leskovec. PinnerSage. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery &amp Data Mining*. ACM, aug 2020.

[36] 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.

[37] Joo-yeong Song and Bongwon Suh. Data augmentation strategies for improving sequential recommender systems. *arXiv preprint arXiv:2203.14037*, 2022.

[38] Zaiqiao Meng, Richard McCreadie, Craig Macdonald, and Iadh Ounis. Exploring data splitting strategies for the evaluation of recommendation models. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pages 681–686, 2020.

[39] Tsvetomila Mihaylova and André FT Martins. Scheduled sampling for transformers. *arXiv preprint arXiv:1906.07651*, 2019.

[40] Aleksandr Petrov and Craig Macdonald. Effective and efficient training for sequential recommendation using recency sampling. In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 81–91, 2022.

[41] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[42] Oren Barkan and Noam Koenigstein. Item2vec: neural item embedding for collaborative filtering. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2016.

[43] Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus Brubaker. Time2vec: Learning a vector representation of time. *arXiv preprint arXiv:1907.05321*, 2019.

[44] Zhiyu Yao, Xinyang Chen, Sinan Wang, Qinyan Dai, Yumeng Li, Tanchao Zhu, and Mingsheng Long. Recommender transformers with behavior pathways, 2022.