



Università degli Studi di Bari Aldo Moro



Gruppo di ricerca Semantic Web Access and Personalization

# Tesi in Metodi per il Ritrovamento dell'Informazione

## **Sustainability of RecSys**

Tesi triennale in Informatica  
Anno accademico 2023/2024

- **Relatore:** Prof. Pasquale Lops
- **Relatore:** Prof. Cataldo Musto
- **Laureando:** Emanuele Fontana

# Indice

<b>1</b>	<b>Abstract</b>	<b>3</b>
<b>2</b>	<b>Sostenibilità</b>	<b>4</b>
2.1	Introduzione alla sostenibilità . . . . .	4
2.2	Sostenibilità ambientale . . . . .	5
2.3	Green AI . . . . .	5
<b>3</b>	<b>Recommender Systems</b>	<b>7</b>
3.1	Introduzione . . . . .	7
3.2	Collaborative Filtering . . . . .	9
3.3	Content-based . . . . .	10
3.4	Approci ibridi . . . . .	11
3.5	Knowledge-based . . . . .	11
3.6	Modelli di raccomandazione a stato dell'arte . . . . .	11
3.7	Recommender Systems e Sustainability . . . . .	11

# 1 Abstract

Lo scopo di questa tesi è quello di analizzare la sostenibilità ambientale dei sistemi di raccomandazione, in particolare di quelli basati su algoritmi di apprendimento automatico.

Si vuole indagare su quello che è il trade-off tra le performance dei modelli di raccomandazione a stato dell'arte e il loro impatto ambientale e vedere se sia possibile ridurre quest'ultimo senza compromettere in modo significativo le performance.

Per fare ciò si è scelto di addestrare alcuni modelli di raccomandazione a stato dell'arte su dataset di dimensioni diverse e cercare di capire il trade-off tra performance e impatto ambientale di ciascuno di essi. Successivamente si è cercato di ridurre l'impatto ambientale di questi modelli lavorando sul criterio di early stopping, cercando di capire se sia possibile utilizzare un criterio di early stopping basato anche sulle emissioni dei modelli. Anche questi esperimenti sono stati condotti su dataset di dimensioni diverse e sui diversi modelli di raccomandazione a stato dell'arte per valutare il trade-off tra performance e impatto ambientale. Successivamente si sono confrontati i risultati ottenuti con quelli precedenti.

I risultati ottenuti mostrano che, in generale, è possibile ridurre l'impatto ambientale dei modelli di raccomandazione senza compromettere in modo significativo le performance dei modelli stessi.

## 2 Sostenibilità

### 2.1 Introduzione alla sostenibilità

La sostenibilità è un concetto che si è diffuso negli ultimi anni, in particolare a partire dagli anni '80, e che ha assunto un ruolo sempre più importante nella società contemporanea. La sostenibilità è un concetto complesso e multidimensionale, che riguarda diversi aspetti della vita umana e dell'ambiente in cui viviamo. In generale, la sostenibilità si riferisce alla capacità di soddisfare i bisogni delle generazioni presenti senza compromettere la capacità delle generazioni future di soddisfare i propri bisogni. Questo significa che la sostenibilità riguarda la gestione responsabile delle risorse naturali, la tutela dell'ambiente, la promozione dello sviluppo economico e sociale, e la garanzia di un futuro migliore per tutti. Un'esempio concreto di impegno per la sostenibilità è l' Agenda 2030 per lo Sviluppo Sostenibile, adottata dalle Nazioni Unite nel 2015. L'Agenda 2030 è un piano d'azione globale che contiene 17 Obiettivi di Sviluppo Sostenibile (SDGs) e 169 obiettivi specifici, che coprono una vasta gamma di temi, tra cui la povertà, la fame, la salute, l'istruzione, l'uguaglianza di genere, l'acqua, l'energia, il clima, l'ambiente, la pace e la giustizia. L'obiettivo dell'Agenda 2030 è quello di promuovere uno sviluppo sostenibile che sia equo, inclusivo e rispettoso dell'ambiente, e di garantire che nessuno venga lasciato indietro.



Figura 1: I 17 Obiettivi di Sviluppo Sostenibile dell'Agenda 2030

In ambito di Intelligence Artificiale, la sostenibilità è un tema di grande rilevanza, in quanto l'uso di tecnologie avanzate come l'AI può avere un impatto significativo sull'ambiente e sulla società. Ad esempio, l'AI richiede una grande quantità di risorse energetiche per funzionare, e può avere un impatto negativo sull'ambiente se non viene utilizzata in modo responsabile. Inoltre, l'AI può avere effetti sociali indesiderati, come la discriminazione e l'esclusione di determinati gruppi di persone. Per questo motivo, è importante che lo sviluppo e l'uso dell'AI siano guidati dai principi della sostenibilità, al fine di garantire che l'AI contribuisca a uno sviluppo sostenibile e equo per tutti.

## 2.2 Sostenibilità ambientale

La sostenibilità ambientale è uno degli aspetti più importanti della sostenibilità, in quanto l'ambiente è la base su cui si fonda la vita umana e la prosperità economica. La sostenibilità ambientale riguarda la gestione responsabile delle risorse naturali, la tutela dell'ambiente e la prevenzione dell'inquinamento e del degrado ambientale. La sostenibilità ambientale si basa su principi come il rispetto per la natura, la conservazione della biodiversità, la riduzione delle emissioni di gas serra e la promozione di energie rinnovabili e pulite. La sostenibilità ambientale è fondamentale per garantire un futuro sostenibile per tutti, e per preservare il pianeta per le generazioni future.

## 2.3 Green AI

In ambito Intelligenza Artificiale e sostenibilità ambientale possiamo distinguere due tipi di AI sostenibile [43]: *Sustainability of AI* e *AI for Sustainability*. Il primo ramo ha come obiettivo quello di misurare la sostenibilità dello sviluppo e dell'uso di modelli AI, ad esempio misurando la *carbon footprint* e l'energia usata per addestrare un modello. Il secondo ramo, invece, si occupa di utilizzare l'AI per affrontare le sfide della sostenibilità, ad esempio sviluppando modelli per la previsione del cambiamento climatico o per la gestione delle risorse naturali.

Con il termine **Green AI** [24] ci si riferisce alla ricerca e allo sviluppo di modelli di intelligenza artificiale che tengano conto del costo computazionale, delle risorse utilizzate e dell'impatto ambientale. Questa si differenzia dalla **Red AI** il cui obiettivo è quello di ottenere modelli sempre più complessi e performanti, senza tenere conto delle risorse utilizzate e dell'impatto ambientale. Dunque la Green AI promuove un approccio in cui si tiene conto del trade-off tra performance e efficienza, cercando di ottenere modelli che siano performanti ma che allo stesso tempo siano sostenibili e rispettosi dell'ambiente. La Green AI tiene dunque conto dell' **efficienza energetica**: i modelli di AI devono essere progettati in modo da utilizzare

meno risorse energetiche possibili, ad esempio riducendo il numero di parametri, la complessità del modello e il numero di operazioni computazionali. Alcuni fattori da tenere in considerazione sono:

- **Hardware:** l'hardware utilizzato per addestrare e eseguire i modelli di AI può avere un impatto significativo sull'efficienza energetica. Ad esempio, l'uso di hardware specializzato come le GPU può ridurre il tempo di addestramento e il consumo energetico rispetto all'uso di CPU tradizionali.
- **Algoritmi:** la scelta degli algoritmi di AI può influenzare l'efficienza energetica dei modelli. Alcuni algoritmi sono più efficienti di altri in termini di consumo energetico e risorse computazionali, e possono essere preferiti per ridurre l'impatto ambientale.
- **Parametri:** il numero di parametri di un modello di AI può influenzare il consumo energetico e le risorse computazionali richieste per addestrare e eseguire il modello. Ridurre il numero di parametri può migliorare l'efficienza energetica del modello.
- **Emissioni di gas serra:** le emissioni di gas serra prodotte durante l'addestramento e l'esecuzione dei modelli di AI possono contribuire al cambiamento climatico e all'inquinamento atmosferico. Ridurre le emissioni di gas serra è un obiettivo importante per la Green AI.

## 3 Recommender Systems

### 3.1 Introduzione

Un sistema di raccomandazione (Recommender System) [17] è un sistema software progettato per suggerire all'utente elementi di interesse, come ad esempio prodotti, servizi, informazioni o contenuti multimediali, in base alle preferenze e ai comportamenti passati dell'utente. I sistemi di raccomandazione sono ampiamente utilizzati in diversi contesti, come ad esempio il commercio elettronico, i social network, i servizi di streaming multimediale e le piattaforme di ricerca e informazione. I sistemi di raccomandazione sono utili per migliorare l'esperienza dell'utente, aumentare la soddisfazione e la fidelizzazione del cliente, e favorire la scoperta di nuovi contenuti e opportunità. Questi sistemi sono basati su algoritmi di apprendimento automatico e intelligenza artificiale, che analizzano i dati relativi alle preferenze e ai comportamenti degli utenti, e generano raccomandazioni personalizzate in base a tali informazioni.



Figura 2: Alcuni famose piattaforme che utilizzano sistemi di raccomandazione

I sistemi di raccomandazione possono essere di diversi tipi, a seconda della tecnica utilizzata per generare le raccomandazioni.

- **Collaborative filtering** [?]
- **Content-based** [?]
- **Knowledge-based** [4]
- **Approci ibridi**: combinano le tecniche precedenti

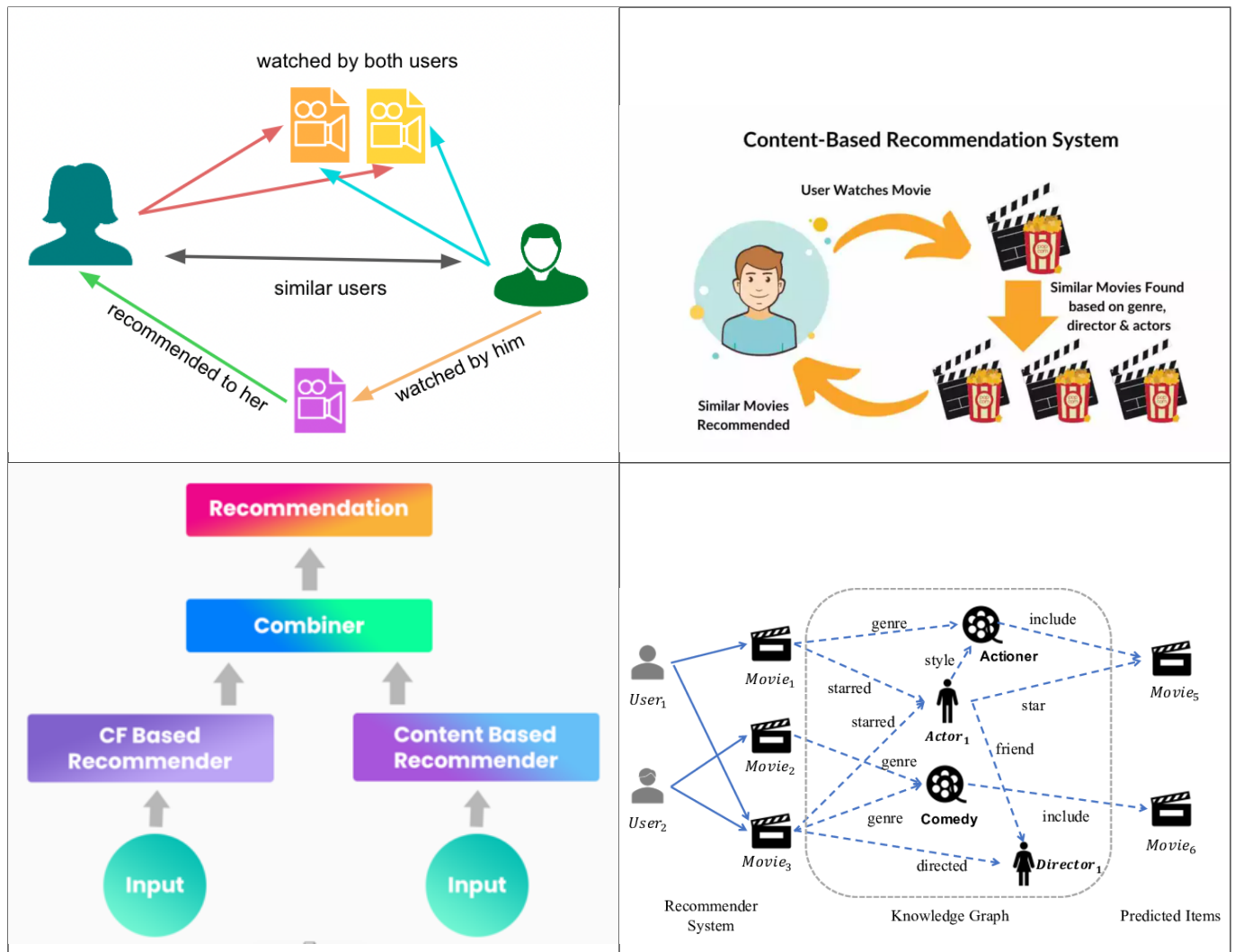


Tabella 1: Tipi di Recommender Systems

Per valutare le prestazioni di un sistema di raccomandazione si possono utilizzare diverse metriche che è possibile riassumere nelle seguenti categorie:

- **Accuracy metrics:** queste metriche valutano la precisione e l'accuratezza delle raccomandazioni generate dal sistema. Alcune delle metriche più comuni sono l'RMSE (Root Mean Squared Error) e il MAE (Mean Absolute Error).
- **Ranking metrics:** queste metriche valutano la qualità dell'ordinamento delle raccomandazioni generate dal sistema. Alcune delle metriche più comuni sono il coefficiente di correlazione di Kendall Tau, il coefficiente di correlazione di Spearman, l'NDCG (Normalized Discounted Cumulative Gain).



- **Diversity metrics:** queste metriche valutano la diversità delle raccomandazioni generate dal sistema. Alcune delle metriche più comuni sono la diversità delle raccomandazioni e la novità delle raccomandazioni.
- **Coverage metrics:** queste metriche valutano la copertura degli elementi raccomandati dal sistema.
- **Classification metrics:** queste metriche valutano la capacità del sistema di classificare correttamente gli elementi in base alle preferenze dell'utente. Alcune delle metriche più comuni sono l'accuracy, la precision, il recall e l'F1-score.

Alcuni tipici problemi che si possono incontrare nella progettazione e nell'implementazione di un sistema di raccomandazione sono:

- **Cold start problem** [23]: il problema del cold start si verifica quando un nuovo utente o un nuovo elemento si registra nel sistema e non ci sono dati sufficienti per generare raccomandazioni personalizzate.
- **Data sparsity problem** [12]: nella maggior parte delle reali applicazioni il numero di item è molto maggiore del numero di item valutati da ciascun utente. Questo porta a una matrice di valutazioni molto sparsa, che rende difficile la generazione di raccomandazioni accurate
- **Vulnerabilità agli attacchi** [20] : i sistemi di raccomandazione possono essere vulnerabili a diversi tipi di attacchi, come ad esempio le recensioni fake (tipico problema degli e-commerce)

## 3.2 Collaborative Filtering

I sistemi di raccomandazione collaborative filtering generano raccomandazioni/filtrano i contenuti basandosi sull' "opinione" di altri utenti.

Con il termine utente ci si riferisce a qualsiasi individuo che inserisca delle valutazioni per gli item presenti nel sistema.

Gli item sono gli oggetti che vengono raccomandati agli utenti (es. film, libri, prodotti, etc.).

L'idea alla base del collaborative filtering è creare una matrice di valutazioni utente-item, in cui ogni cella della matrice rappresenta la valutazione di un utente per un item.

Le valutazioni sono in genere numeriche, e possono essere espresse in termini di rating (es. da 1 a 5 stelle) o di preferenze (es. like/dislike).

Esistono principalmente due tipi di collaborative filtering:

- **User-based collaborative filtering:** in questo approccio, le raccomandazioni vengono generate confrontando le preferenze dell'utente con quelle degli altri utenti. In particolare, si calcola la similarità tra l'utente target e gli altri utenti, e si generano raccomandazioni basate sulle preferenze degli utenti più simili all'utente target. Due utenti sono ritenuti simili se hanno uno stile di valutazione simile per gli item (cioè valutano gli stessi item in modo simile)
- **Item-based collaborative filtering:** in questo approccio, le raccomandazioni vengono generate confrontando le preferenze degli utenti per gli item. In particolare, si calcola la similarità tra gli item, e si generano raccomandazioni basate sugli item più simili a quelli valutati positivamente dall'utente target. Due item sono ritenuti simili se vengono valutati in modo simile dagli stessi utenti.

I principali vantaggi del collaborative filtering sono la sua semplicità e la sua capacità di generare raccomandazioni personalizzate senza la necessità di dover conoscere le caratteristiche degli item (es. la durata di un film, il genere di un libro, etc.). Tuttavia, il collaborative filtering può soffrire di problemi come il cold start (per un nuovo utente e per un nuovo item), la vulnerabilità agli attacchi come le recensioni fake e la scarsa spiegabilità delle raccomandazioni generate.

### 3.3 Content-based

I sistemi di raccomandazione content-based generano raccomandazioni basate sul contenuto degli elementi e sulle preferenze dell'utente. Questi sistemi analizzano le caratteristiche degli elementi e le preferenze dell'utente, e generano raccomandazioni in base alla somiglianza tra gli elementi e le preferenze dell'utente. Per preferenze dell'utente si intendono le caratteristiche degli elementi che l'utente ha valutato in passato. Un sistema di raccomandazione content-based è composto da tre componenti principali:

- **Profile learner:** Questa componente colleziona i dati relativi alle preferenze dell'utente e cerca di generalizzarle per creare un profilo dell'utente. Spesso questa generalizzazione avviene mediante tecniche di apprendimento automatico.
- **Content analyzer:** Questa componente ha come scopo quello di estrarre le caratteristiche degli item. Quando le descrizioni non sono strutturate (es. testo) è necessaria una fase di pre-processing per estrarre le caratteristiche rilevanti.
- **Filtering Component:** Questa componente ha come scopo quello di generare raccomandazioni personalizzate in base al profilo dell'utente e alle caratteristiche degli item. In particolare, si calcola

la somiglianza tra il profilo dell'utente e le caratteristiche degli item, e si generano raccomandazioni basate su questa somiglianza.

I principali vantaggi sono l'indipendenza dal comportamento degli altri utenti e la capacità di generare raccomandazioni personalizzate anche per nuovi item. Inoltre c'è una maggiore spiegabilità delle raccomandazioni generate. Tuttavia, i sistemi di raccomandazione content-based possono soffrire di problemi come la scarsa diversità delle raccomandazioni e la difficoltà di estrarre le caratteristiche rilevanti degli item. Rimane comunque il problema di cold-start per un nuovo utente.

### 3.4 Knowledge-based

I sistemi di raccomandazione knowledge-based generano raccomandazioni basandosi su conoscenza semantica, rappresentata mediante i knowledge-graph. I knowledge-graph sono grafi in cui i nodi rappresentano concetti e le relazioni tra i concetti, e gli archi rappresentano le relazioni tra i concetti. I sistemi di raccomandazione knowledge-based utilizzano i knowledge-graph per rappresentare le caratteristiche degli item e le preferenze dell'utente, e generare raccomandazioni basate su questa rappresentazione. In particolare, si utilizzano tecniche di reasoning per inferire nuove conoscenze a partire dalle conoscenze esistenti, e generare raccomandazioni basate su queste nuove conoscenze. I principali vantaggi dei sistemi di raccomandazione knowledge-based l'assenza del problema di cold-start. Tuttavia, i sistemi di raccomandazione knowledge-based possono soffrire di problemi come la scarsa spiegabilità delle raccomandazioni generate e la difficoltà di rappresentare la conoscenza semantica in modo accurato e completo. Inoltre, i sistemi di raccomandazione knowledge-based possono richiedere una quantità significativa di risorse computazionali per generare raccomandazioni accurate e rilevanti.

### 3.5 Modelli di raccomandazione a stato dell'arte

### 3.6 Recommender Systems e Sustainability

I sistemi di raccomandazione, così come tutti gli altri modelli di AI, possono essere utilizzati per promuovere la sostenibilità in tutti i suoi punti, ad esempio cercando di adempiere agli obiettivi dell'Agenda 2030 [7]. In ambito *energia pulita e riduzione delle emissioni* viene suggerito come i sistemi di raccomandazione possano essere utilizzati per promuovere l'adozione di comportamenti sostenibili, ad esempio suggerendo all'utente di utilizzare mezzi di trasporto pubblici o condivisi, di ridurre il consumo di energia elettrica

o di acquistare prodotti sostenibili. Inoltre, i sistemi di raccomandazione possono essere utilizzati per promuovere l'adozione di energie rinnovabili e la riduzione delle emissioni di gas serra. In questo caso si parla dunque di sistemi di raccomandazione i quali cercano di promuovere comportamenti sostenibili, ma a loro volta per essere addestrati richiedono grandi quantità di dati e di risorse computazionali, che possono avere un impatto negativo sull'ambiente.

Una soluzione dunque può essere quella di addestrare i modelli di raccomandazione in modo sostenibile, senza però perdere di performance.

Tracciare le emissioni degli algoritmi di raccomandazione e cercare di prevederle è molto importante quando si parla di sviluppo sostenibile in campo RecSys. Ancora oggi si tende a trascurare l'impatto ambientale di un'attività e, in questo ambito, si è molto propensi nell'utilizzare dei modelli molto complessi e pesanti che richiedono molte risorse per essere addestrati ed eseguiti per ottenere delle buone performance. Spesso, però, modelli molto più leggeri e semplici riescono a ottenere delle performance molto simili (se non superiori) a modelli più complessi e il tutto con un impatto ambientale decisamente minore. Ad oggi il carbon dioxide equivalent ( $\text{CO}_2\text{eq}$ ) è il principale indicatore utilizzato da governi e enti per misurare l'impatto ambientale di un'attività. Il  $\text{CO}_2\text{eq}$  è un'unità di misura che esprime l'equivalente in  $\text{CO}_2$  di tutti i gas serra emessi da un'attività, in modo da poter confrontare l'impatto ambientale di attività diverse. Una strategia comune per calcolare il  $\text{CO}_2\text{eq}$  è quella di moltiplicare tra loro il **carbon intensity(CI)** e l'**energia consumata(PC)** dall'attività (nel nostro caso l'esecuzione di algoritmi).

$$emission = CI \cdot PC$$

In particolare i valori di CI dipendono dalle diverse fonti di energia utilizzate durante la computazione (es. energia solare, energia eolica, etc.). Se  $s$  è la fonte di energia,  $e_s$  sono le emissioni per KW/h di energia e  $p_s$  è la percentuale di energia prodotta dalla fonte  $s$ , allora il CI è dato da:

$$CI = \sum_{s \in S} e_s \cdot p_s$$

## Riferimenti bibliografici

- [1] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9), 2018.
- [2] Fabio Aiolli. Efficient top-n recommendation for very large scale binary rated datasets. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 273280, New York, NY, USA, 2013. Association for Computing Machinery.
- [3] Ting Bai, Ji-Rong Wen, Jun Zhang, and Wayne Xin Zhao. A neural collaborative filtering model with interaction-based neighborhood. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, page 19791982, New York, NY, USA, 2017. Association for Computing Machinery.
- [4] Robin Burke. Knowledge-based recommender systems. *Encyclopedia of library and information systems*, 69(Supplement 32):175–186, 2000.
- [5] Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu, and Tat-Seng Chua. Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In *The World Wide Web Conference*, WWW '19, page 151161, New York, NY, USA, 2019. Association for Computing Machinery.
- [6] Chong Chen, Min Zhang, Yongfeng Zhang, Yiqun Liu, and Shaoping Ma. Efficient neural matrix factorization without sampling for recommendation. *ACM Trans. Inf. Syst.*, 38(2), jan 2020.
- [7] Alexander Felfernig, Manfred Wundara, Thi Ngoc Trang Tran, Seda Polat-Erdeniz, Sebastian Lubos, Merfat El Mansi, Damian Garber, and Viet-Man Le. Recommender systems for sustainability: overview and research issues. *Frontiers in Big Data*, 6, 2023.
- [8] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 639648, New York, NY, USA, 2020. Association for Computing Machinery.
- [9] Xiangnan He, Xiaoyu Du, Xiang Wang, Feng Tian, Jinhui Tang, and Tat-Seng Chua. Outer product-based neural collaborative filtering. *arXiv preprint arXiv:1808.03912*, 2018.

- [10] Xiangnan He, Zhankui He, Jingkuan Song, Zhenguang Liu, Yu-Gang Jiang, and Tat-Seng Chua. Nais: Neural attentive item similarity model for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 30(12):23542366, December 2018.
- [11] 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, WWW '17*, page 173182, Republic and Canton of Geneva, CHE, 2017. International World Wide Web Conferences Steering Committee.
- [12] Zan Huang, Hsinchun Chen, and Daniel Zeng. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Trans. Inf. Syst.*, 22(1):116142, jan 2004.
- [13] Santosh Kabbur, Xia Ning, and George Karypis. Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, page 659667, New York, NY, USA, 2013. Association for Computing Machinery.
- [14] 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, WWW '18*, pages 689–698, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.
- [15] 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, WWW '18*, page 689698, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.
- [16] Zihan Lin, Changxin Tian, Yupeng Hou, and Wayne Xin Zhao. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *Proceedings of the ACM Web Conference 2022, WWW 22*. ACM, April 2022.
- [17] Linyuan Lü, Matú Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, and Tao Zhou. Recommender systems. *Physics Reports*, 519(1):1–49, 2012. Recommender Systems.
- [18] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. *Learning disentangled representations for recommendation*. Curran Associates Inc., Red Hook, NY, USA, 2019.

- [19] Kelong Mao, Jieming Zhu, Jinpeng Wang, Quanyu Dai, Zhenhua Dong, Xi Xiao, and Xiuqiang He. Simplex: A simple and strong baseline for collaborative filtering. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 12431252, New York, NY, USA, 2021. Association for Computing Machinery.
- [20] Bamshad Mobasher, Robin Burke, Runa Bhaumik, and Chad Williams. Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.*, 7(4):23es, oct 2007.
- [21] 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, 2011.
- [22] 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*, UAI '09, page 452461, Arlington, Virginia, USA, 2009. AUAI Press.
- [23] 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*, SIGIR '02, page 253260, New York, NY, USA, 2002. Association for Computing Machinery.
- [24] Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. Green ai. *Commun. ACM*, 63(12):5463, nov 2020.
- [25] Ilya Shenbin, Anton Alekseev, Elena Tutubalina, Valentin Malykh, and Sergey I. Nikolenko. Recvae: A new variational autoencoder for top-n recommendations with implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, WSDM '20, page 528536, New York, NY, USA, 2020. Association for Computing Machinery.
- [26] Giuseppe Spillo, Allegra De Filippo, Cataldo Musto, Michela Milano, and Giovanni Semeraro. Towards sustainability-aware recommender systems: analyzing the trade-off between algorithms performance and carbon footprint. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 856–862, 2023.

- [27] Harald Steck. Embarrassingly shallow autoencoders for sparse data. In *The World Wide Web Conference*, WWW '19, page 32513257, New York, NY, USA, 2019. Association for Computing Machinery.
- [28] Harald Steck, Maria Dimakopoulou, Nickolai Riabov, and Tony Jebara. Admm slim: Sparse recommendations for many users. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, WSDM '20, page 555563, New York, NY, USA, 2020. Association for Computing Machinery.
- [29] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, page 10671077, Republic and Canton of Geneva, CHE, 2015. International World Wide Web Conferences Steering Committee.
- [30] Rianne van den Berg, Thomas N. Kipf, and Max Welling. Graph convolutional matrix completion, 2017.
- [31] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 417426, New York, NY, USA, 2018. Association for Computing Machinery.
- [32] Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems, 2019.
- [33] Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Multi-task feature learning for knowledge graph enhanced recommendation, 2019.
- [34] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. Knowledge graph convolutional networks for recommender systems. In *The World Wide Web Conference*, WWW '19, page 33073313, New York, NY, USA, 2019. Association for Computing Machinery.
- [35] Wenjie Wang, Yiyang Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 832841, New York, NY, USA, 2023. Association for Computing Machinery.



- [36] Wenjie Wang, Yiyang Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 832841, New York, NY, USA, 2023. Association for Computing Machinery.
- [37] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 165174, New York, NY, USA, 2019. Association for Computing Machinery.
- [38] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference 2021*, WWW '21, page 878887, New York, NY, USA, 2021. Association for Computing Machinery.
- [39] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. Disentangled graph collaborative filtering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 10011010, New York, NY, USA, 2020. Association for Computing Machinery.
- [40] Ga Wu, Maksims Volkovs, Chee Loong Soon, Scott Sanner, and Himanshu Rai. Noise contrastive estimation for one-class collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 135144, New York, NY, USA, 2019. Association for Computing Machinery.
- [41] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. Self-supervised graph learning for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 21. ACM, July 2021.
- [42] Yao Wu, Christopher DuBois, Alice X. Zheng, and Martin Ester. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, page 153162, New York, NY, USA, 2016. Association for Computing Machinery.
- [43] Aimee Wynsberghe. Sustainable ai: Ai for sustainability and the sustainability of ai. *AI and Ethics*, 1, 02 2021.

- 
- [44] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. Deep matrix factorization models for recommender systems. In *IJCAI*, volume 17, pages 3203–3209. Melbourne, Australia, 2017.
- [45] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 353362, New York, NY, USA, 2016. Association for Computing Machinery.
- [46] Lei Zheng, Chun-Ta Lu, Fei Jiang, Jiawei Zhang, and Philip S Yu. Spectral collaborative filtering. In *Proceedings of the 12th ACM conference on recommender systems*, pages 311–319, 2018.