

Uncertainty-aware Consistency Learning for Cold-Start Item Recommendation

Taichi Liu
liutq20@mails.tsinghua.edu.cn
Tsinghua University

Chen Gao*
chgao96@gmail.com
Tsinghua University
Huawei Noah's Ark Lab

Zhenyu Wang
wangzy20@mails.tsinghua.edu.cn
Tsinghua University

Dong Li
lidong106@huawei.com
Huawei Noah Ark's Lab

Jianye Hao
haojianye@126.com
Huawei Noah's Ark Lab

Depeng Jin
jindp@tsinghua.edu.cn
Tsinghua University

Yong Li
liyong07@tsinghua.edu.cn
Tsinghua University

ABSTRACT

Graph Neural Network (GNN)-based models have become the mainstream approach for recommender systems. Despite the effectiveness, they are still suffering from the cold-start problem, *i.e.*, recommend for few-interaction items. Existing GNN-based recommendation models to address the cold-start problem mainly focus on utilizing auxiliary features of users and items, leaving the user-item interactions under-utilized. However, embeddings distributions of cold and warm items are still largely different, since cold items' embeddings are learned from lower-popularity interactions, while warm items' embeddings are from higher-popularity interactions. Thus, there is a *seesaw phenomenon*, where the recommendation performance for the cold and warm items cannot be improved simultaneously. To this end, we proposed a Uncertainty-aware Consistency learning framework for Cold-start item recommendation (shorten as UCC) solely based on user-item interactions. Under this framework, we train the teacher model (generator) and student model (recommender) with consistency learning, to ensure the cold items with additionally generated low-uncertainty interactions can have similar distribution with the warm items. Therefore, the proposed framework improves the recommendation of cold and warm items at the same time, without hurting any one of them. Extensive experiments on benchmark datasets demonstrate that our proposed method significantly outperforms state-of-the-art methods on both warm and cold items, with an average performance improvement of 27.6%.

CCS CONCEPTS

• **Information systems** → **Recommender systems.**

*Chen Gao is the Corresponding Author.

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 the author(s) 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.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9408-6/23/07...\$15.00
<https://doi.org/10.1145/3539618.3592078>

KEYWORDS

Recommender System; Cold-Start Item; Graph Neural Networks

ACM Reference Format:

Taichi Liu, Chen Gao, Zhenyu Wang, Dong Li, Jianye Hao, Depeng Jin, and Yong Li. 2023. Uncertainty-aware Consistency Learning for Cold-Start Item Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*, July 23–27, 2023, Taipei, Taiwan. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3539618.3592078>

1 INTRODUCTION

Recommender systems have become a fundamental service for filtering information by learning from collected user behavioral data and inferring users' personalized demands. Currently, GNN-based models, with their strengths in learning from structured data and capturing high-order similarity, have become state-of-the-art approaches in various recommendation tasks [9]. GNNs have a strong ability to incorporate information from the neighboring nodes, helping to understand the local context of nodes. However, it leads to the poor performance for the cold-start recommendation [1, 4, 14, 21, 26, 27, 29], which occurs when a new item node is added to the graph. Specifically, there are not enough neighbors for the new item due to few user-item interactions. The cold-start recommendation problem can be particularly challenging in scenarios where the user-item graph rapidly evolves, such as short-video recommendation [15].

To address this challenge, existing models [5, 8, 20, 21, 30] have proposed various techniques for incorporating additional information into GNNs, such as social networks, node attributes, and other metadata, which can partly improve the performance for cold-start items. However, cold items' embeddings are still continuously learned from fewer interactions, while warm items' embeddings are still mainly learned from adequate interactions [2, 3], which leads to the issue that the distributions of embeddings of cold and warm items are significantly different. That is, the collected interaction of cold items cannot reflect the real item characteristics due to less exposure, thus resulting in the *seesaw phenomenon*, *i.e.*, improves the recommendation of either the cold or warm items will hurt the other [6, 28].

To this end, we proposed a framework for bridging the distribution gap between cold items and warm items in the cold-start recommendation, with two key designs included: a) **Uncertainty-aware Interaction Generation**, b) **Graph-based Teacher-Student Consistency Learning**. In particular, we first introduce low-uncertainty interactions and take use of generated interactions for embedding learning, to address the distribution difference which causes the seesaw phenomenon. We then develop a teacher-student training paradigm that improves the robustness [16, 24] in embedding learning for cold and warm items. Specifically, we propose a contrastive loss to keep the consistency of item embeddings before and after the interaction generation. We further fine-tune the student model on the recommendation task, maintaining the consistency between the teacher model and student model. Our contributions can be summarized as follows:

- We propose approach the cold-start problem from the novel perspective of addressing the distribution difference between cold and warm items, by proposing a general UCC framework.
- Under the framework, we first address the distribution gap between cold and warm items through our uncertainty-aware interaction generation, and keep both item-level and model-level similarity with consistency learning, which can improve model's robustness.
- We conduct extensive experiments to demonstrate the effectiveness of our framework. The results shown an average improvement of 27.6% in ranking metrics compared to the baseline method on two widely-used benchmark datasets.

2 PROBLEM FORMULATION

Notations. The common GNN-based model paradigm can be formulated as follows: Let $\mathcal{U} = \{u\}$, $\mathcal{I} = \{i\}$ denote the set of users and items respectively. The interactions of user and item can be regard as $\mathcal{O} = \{(u, i^+) \mid u \in \mathcal{U}, i^+ \in \mathcal{I}\}$, where each pair represents each observed feedback.

Input: Most GNN-based models treat user-item interactions as a bipartite graph, where the node set is $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$ and the edge set is $\mathcal{G} = (\mathcal{V}, \mathcal{O}^+)$. In the training process of graph representation learning, we use $\mathbf{E}_u = [e_{u_1}, \dots, e_{u_M}] \in \mathbb{R}^{M \times D}$, $\mathbf{E}_i = [e_{i_1}, \dots, e_{i_N}] \in \mathbb{R}^{N \times D}$ to denote the user embedding and item embedding, where D is the dimension size of embedding, M, N are the number of users and items respectively.

Output: Recommender models estimate the relations of unobserved user-item pairs through the dot products of their embeddings [10, 12, 13]. Specifically, the scores $\{s_{mn}\}$ of a given user m and a given item n , where s_{mn} is calculated as:

$$s_{mn} = e_{u_m} e_{i_n}^T, \quad (1)$$

then rank items by their scores - a high score means that the user prefers. The top- k items from the ranking list are adopted into a candidate set and recommended to the user.

3 METHODOLOGY

3.1 Uncertainty-aware Interaction Generation

For the cold-start recommendation, one main reason for the limited performance is the different distribution between cold items and warm items caused by popularity bias. Hence, we propose to

generate similar distributed interactions for cold items as warm items, and we need to select accurate and reliable interactions from unobserved interactions.

However, since totally unbiased and accurate interactions for the original data used are unavailable, especially for cold items, it is hard to discriminate whether an interaction is reliable. We thus propose to estimate the uncertainty degree of each user-item interaction, and select low-uncertainty interactions for each item. We estimate the uncertainty of the particular interaction between the user u_m and item i_n by their similarity. We apply the cosine distance to calculate the similarity:

$$d_{mn} = \frac{|e_{u_m} e_{i_n}^T|}{\|e_{u_m}\| \|e_{i_n}\|}, \quad (2)$$

With the pre-trained recommender, we have the ranking scores $\{s_{mn}\}_{n=1}^N$ of each item i_n for all users. The average of all ranking scores measures the overall uncertainty degree of the item i_n as follows:

$$\bar{s}_n = \frac{1}{M} \sum_{m=1}^M s_{mn}, \quad (3)$$

where \bar{s}_n provides a description of the overall interaction uncertainty of the item i_n . We aim to force the cold items with generated interactions to have a similar distribution as the warm items to improve the recommendation performance for both cold and warm items. A small d_{nm} interactions are regarded as uncertain ones and are filtered out in the generation phase. This selection strategy based on the interaction uncertainty guides the generator on more certain interactions to mitigate the popularity bias in the cold-start problem. The selection is as follows:

$$\widehat{\mathcal{O}}_n = I(d_{mn} > \alpha \bar{s}_n), \quad (4)$$

where α is a pre-defined parameter and I is the indicator function.

3.2 Teacher-Student Consistency Learning

The consistency learning for our framework consists of two parts:

3.2.1 Item-level consistency learning. We adopt the contrastive loss for the item embeddings between pre-generation and post-generation. Inspired by [19], we adopt two augmentations - strong augmentation and weak augmentation. Specifically, the weak augmentation drops out the edges in the graph with a dropout ratio ρ , which can be formulated as follows:

$$\mathcal{G}^w = (\mathcal{V}, \mathcal{M} \cdot \mathcal{O}^+), \quad (5)$$

where $\mathcal{M} \in \{0, 1\}^{|\mathcal{O}^+|}$ is the masking vector. The strong augmentation based on generated labels adds more edges to the graph, which can be formulated as follows,

$$\mathcal{G}^s = (\mathcal{V}, \mathcal{O}^+ + \widehat{\mathcal{O}}), \quad (6)$$

Two augmentation operations on the graph generates two different views of each node. For the item node i , we denote its two different views as z'_i, z''_i (on weak and strong graphs respectively). We perform a consistency regularization for these different views, to encourage the similarity between different views of the same

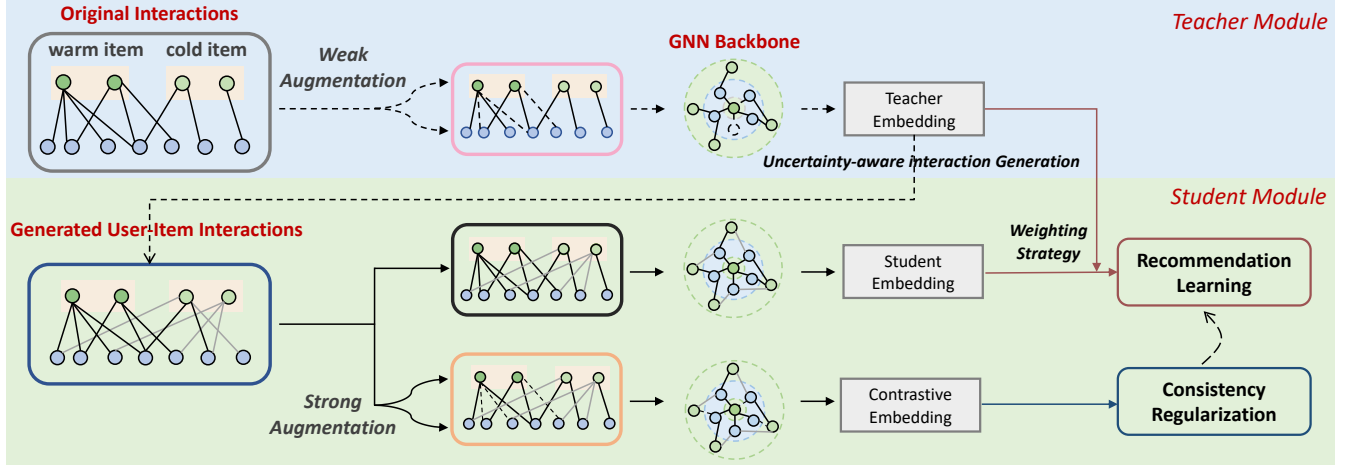


Figure 1: The overall framework of UCC. A teacher model (the upper blue frame) is first trained under weak augmentation regularization. Then interactions are generated with uncertainty-aware selection process. Finally the generated interactions serve for the student model training (the lower green frame). Strong augmentation based consistency regularization and integrated teacher representation increase the robustness of the student model.

node. To implement the consistency regularization, the contrastive loss [7] is as follows,

$$\mathcal{L}_{\text{cr, item}} = \sum_{i \in \mathcal{I}} -\log \frac{\exp(\text{sim}(z'_i, z''_i)/\tau)}{\sum_{j \in \mathcal{I}} \exp(\text{sim}(z'_i, z''_j)/\tau)}, \quad (7)$$

where $\mathcal{L}_{\text{cr, item}}$ denotes the item-side consistency regularization loss for the teacher and the student model, respectively. Here $\text{sim}(\cdot, \cdot)$ denotes the similarity function, for which we use cosine similarity, and τ is a pre-defined hyper-parameter. The user-side consistency regularization loss $\mathcal{L}_{\text{cr, user}}$ can be similarly calculated, and we omit it due to the limit of space. The final consistency loss $\mathcal{L}_{\text{cr}} = \mathcal{L}_{\text{cr, item}} + \mathcal{L}_{\text{cr, user}}$ serves for consistency regularization. For the recommendation loss we use Bayesian personalized ranking (BPR) [17] loss to optimize the representation learning, which can be formulated as follows,

$$\mathcal{L}_{\text{rec}} = \sum_{(u, i^+, i^-) \in \mathcal{O}} -\ln \sigma(\hat{y}_{ui^+} - \hat{y}_{ui^-}) + \lambda \|\Theta\|_2^2, \quad (8)$$

where (u, i^+) , (u, i^-) represent observed/unobserved interaction pairs and $\|\Theta\|_2^2$ is L2-regularization of model's parameters. Then our final loss of the whole framework is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \mu \mathcal{L}_{\text{cr}}, \quad (9)$$

where μ is a hyper-parameter.

3.2.2 Model-level consistency learning. To maintain the consistency between the teacher model and student model, We denote the embedding of the student model's embedding as \mathbf{E}^s . After each iteration, we propose to accumulate the teacher embedding into the student embedding:

$$\mathbf{E}^s \leftarrow \gamma \mathbf{E}^s + (1 - \gamma) \mathbf{E}^t, \quad (10)$$

where γ denotes a momentum term that controls the impact of the teacher model. Here the accumulation step includes both user embeddings and item embeddings.

Table 1: Performance comparison on benchmark datasets.

Dataset	Yelp		Amazon-Book	
Metric	Recall	NDCG	Recall	NDCG
LightGCN	0.0639	0.0515	0.0410	0.0318
IRBPR	0.0538	0.0438	0.0325	0.0246
NB	0.0640	0.0526	0.0425	0.0328
ADT	<u>0.0691</u>	0.0388	0.0335	0.0176
SGL	0.0673	<u>0.0554</u>	<u>0.0477</u>	<u>0.0378</u>
UCC	0.0713	0.0587	0.0523	0.0410
Improv.	+11.6%	+14.0%	+27.6%	+28.9%

4 EXPERIMENTS

4.1 Experimental Setting

Datasets. We conduct experiments on benchmark datasets: Yelp and Amazon-Book, following the same 10-core setting as [11, 12]. We split all user-item interactions into training, validation, and testing set with the ratio of 7:1:2, evaluating the top-K recommendation performance with two widely-used metrics Recall@K and NDCG@K where $K = 20$, following [11].

Baselines. We select LightGCN[11] as the backbone of our GNN-based model. Since we mainly focus on modeling user-item interactions but not the features of items, we compare our model with state-of-the-art recommendation models which can be classified into two classes: generative model NB[18] and denosing model IRBPR[23], ADT[22], SGL[25] for user-item interactions.

4.2 Recommendation Performance Comparison

We evaluate the model from two aspects, overall recommendation performance and cold-start recommendation performance.

4.2.1 Overall Performance. The details of the comparison are listed in Table 1. The following points are observed:

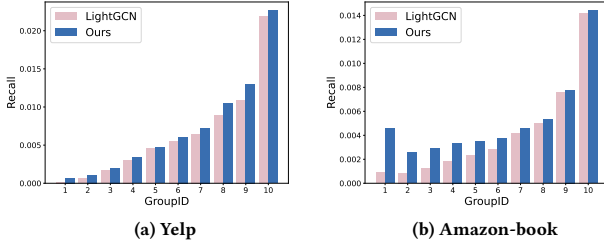


Figure 2: Performance comparison over item groups.

- Our UCC outperforms previous models significantly on all datasets. On the Yelp dataset, our method outperforming recall of LightGCN by 11.6%. Compared to the state-of-the-art graph based method SGL, our method also obtains a higher recall of SGL by 6%. The improvement is consistent for Amazon-Book. This demonstrates the effectiveness of our uncertainty-aware consistency learning.

- Our method improves Recall@20 significantly more on the Amazon-Book dataset - 27.6% compared to LightGCN and 9.6% higher than SGL. The reason is that the interactions for cold items in amazon-book are more biased and sparse. This validates that our method targets at the cold-start problem and well alleviates the restriction of different distribution between cold items and warm items.

4.2.2 Cold-Start Performance. We further compare our methods with LightGCN on cold and warm items separately to illustrate its effectiveness for cold-start recommendation. We split items into ten groups based on the popularity, meanwhile keeping the total number of interactions of each group the same. We treat the last two groups as cold-start items. The larger the GroupID is, the warmer the items are. We evaluate our method on different groups and plot the Recall@20 in Figure 2. From the results, we notice that our method outperforms LightGCN on all groups, no matter whether the items are warm or cold. In comparison, many current methods improve the performance of cold items at the cost of warm item accuracy. Also, our method improves the recall of LightGCN more prominently on cold items. On the Yelp dataset, our method obtains nearly 7 times compared to recall of LightGCN. The most significant improvement comes from the groupid 1 items of Amazon-Book. In this group, the recall of amazon-book obtains is 0.0009. In contrast, the recall of our method obtains 0.0045, improved by 400%. This strongly demonstrates the effectiveness of our method for cold-start recommendation and the problem of the seesaw phenomenon.

4.3 Ablation Study of UCC

In this section, we perform ablation studies to evaluate each component of UCC, which consists of two parts: Uncertainty-aware Interaction Generation and Teacher-Student Consistency Learning.

4.3.1 Uncertainty-aware interaction generation. We calculate the average number of generated interactions for different item groups. We find the number of generated interactions is adaptive for different items. For cold items, the average number of produced interactions is about 3.5, which is about 1.0 for warm items in comparison. Since the generated low-uncertainty interactions contribute more for cold items, the uncertainty guided strategy well alleviating the

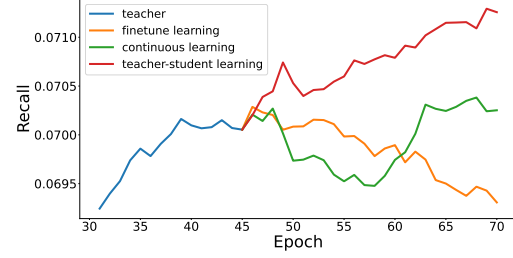


Figure 3: Study of teacher-student learning.

different distribution between warm items and cold items. We also notice that the simple usage of item-side generated interactions helps improves the performance. The performance increase indicates that our item-side adaptive generated interactions can indeed alleviate the cold-start problem. User-side generated interactions, in comparison, cannot improve the recommendation ability of the model - even lowering the Recall@20 from 0.0673 to 0.0671. This is because user-side generated interactions are biased and exacerbate the distribution between cold and warm items. This demonstrates the superiority of our uncertainty-aware interaction generation.

4.3.2 Teacher-Student Consistency Learning. We show the effectiveness of our teacher-student learning approach. We design two comparing experiments: 1) Finetune learning: We train the student representations with generated interactions directly from the teacher model embeddings without accumulating teacher representations. 2) Continuous learning: We train the student embeddings with accumulating with teacher embeddings without using generated interactions. We plot the results in Figure 3. We observe that our teacher-student learning approach obtains a higher performance. The superior comes from two aspects. Compared with finetune learning, our student model maintains more information from the teacher model. The information within the teacher embeddings alleviates the negative effect of noise from generated interactions. Compared with continuous learning, we provide more interactions for cold items, thus mitigating the cold-start problem. Through the proposed teacher-student learning framework, similar distributions of information and data are accumulated together, thus helping obtain a better recommendation performance.

5 CONCLUSION

In this paper, we identify that the main limitation comes from the interaction-distribution difference between cold and warm items. To address it, we propose an uncertainty-aware consistency learning framework. The extraordinary performance on benchmark datasets and its easy-to-use property make our framework practical in both academia and industry.

6 ACKNOWLEDGEMENT

This work was supported by the National Key Research and Development Program of China under 2022YFB3104702, the National Natural Science Foundation of China under 62272262, 61972223, U1936217, and U20B2060, the Fellowship of China Postdoctoral Science Foundation under 2021TQ0027 and 2022M710006.

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 Abdollahpour, Robin Burke, and Bamshad Mobasher. 2019. Managing popularity bias in recommender systems with personalized re-ranking. *arXiv preprint arXiv:1901.07555* (2019).
- [3] Himan Abdollahpour, Masoud Mansoury, Robin Burke, Bamshad Mobasher, and Edward Malthouse. 2021. User-centered evaluation of popularity bias in recommender systems. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*. 119–129.
- [4] Rocio Cañameres and Pablo Castells. 2018. Should I follow the crowd? A probabilistic analysis of the effectiveness of popularity in recommender systems. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 415–424.
- [5] Hao Chen, Zefan Wang, Feiran Huang, Xiao Huang, Yue Xu, Yishi Lin, Peng He, and Zhoujun Li. 2022. Generative adversarial framework for cold-start item recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2565–2571.
- [6] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240* (2020).
- [7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *ICML*.
- [8] Jing Du, Zesheng Ye, Lina Yao, Bin Guo, and Zhiwen Yu. 2022. Socially-aware Dual Contrastive Learning for Cold-Start Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1927–1932.
- [9] Chen Gao, Yu Zheng, Nian Li, Yinfeng Li, Yingrong Qin, Jinghua Piao, Yuhuan Quan, Jianxin Chang, Depeng Jin, Xiangnan He, et al. 2023. A survey of graph neural networks for recommender systems: challenges, methods, and directions. *ACM Transactions on Recommender Systems* 1, 1 (2023), 1–51.
- [10] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.
- [11] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*.
- [12] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*.
- [13] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *Proceedings of the 26th international conference on world wide web*. 193–201.
- [14] Liang Hu, Songlei Jian, Longbing Cao, Zhiping Gu, Qingkui Chen, and Artak Amirbekyan. 2019. Hers: Modeling influential contexts with heterogeneous relations for sparse and cold-start recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3830–3837.
- [15] Chen Ling, Jeremy Blackburn, Emiliano De Cristofaro, and Gianluca Stringhini. 2022. Slapping Cats, Bopping Heads, and Oreo Shakes: Understanding Indicators of Virality in TikTok Short Videos. In *14th ACM Web Science Conference 2022*. 164–173.
- [16] Seung-Taek Park, David Pennock, Omid Madani, Nathan Good, and Dennis DeCoste. 2006. Naïve filterbots for robust cold-start recommendations. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 699–705.
- [17] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).
- [18] Yuta Saito. 2020. Asymmetric tri-training for debiasing missing-not-at-random explicit feedback. In *SIGIR*.
- [19] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS* (2020).
- [20] Riku Togashi, Mayu Otani, and Shin'ichi Satoh. 2021. Alleviating cold-start problems in recommendation through pseudo-labelling over knowledge graph. In *Proceedings of the 14th ACM international conference on web search and data mining*. 931–939.
- [21] Maksims Volkovs, Guangwei Yu, and Tomi Poutanen. 2017. Dropoutnet: Addressing cold start in recommender systems. *NeurIPS* (2017).
- [22] Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. 2021. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*. 373–381.
- [23] Zitai Wang, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Implicit Feedbacks are Not Always Favorable: Iterative Relabeled One-Class Collaborative Filtering against Noisy Interactions. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3070–3078.
- [24] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. 2021. Contrastive learning for cold-start recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 5382–5390.
- [25] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *SIGIR*.
- [26] Hongzhi Yin, Bin Cui, Jing Li, Junjie Yao, and Chen Chen. 2012. Challenging the long tail recommendation. *arXiv preprint arXiv:1205.6700* (2012).
- [27] Jiani Zhang, Xingjian Shi, Shenglin Zhao, and Irwin King. 2019. Star-gcn: Stacked and reconstructed graph convolutional networks for recommender systems. *arXiv preprint arXiv:1905.13129* (2019).
- [28] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal intervention for leveraging popularity bias in recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 11–20.
- [29] Minghao Zhao, Le Wu, Yile Liang, Lei Chen, Jian Zhang, Qilin Deng, Kai Wang, Xudong Shen, Tangjie Lv, and Runze Wu. 2022. Investigating Accuracy-Novelty Performance for Graph-based Collaborative Filtering. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 50–59.
- [30] Yongchun Zhu, Ruobing Xie, Fuzhen Zhuang, Kaikai Ge, Ying Sun, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to warm up cold item embeddings for cold-start recommendation with meta scaling and shifting networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1167–1176.