Post-Training Embedding Enhancement for Long-Tail Recommendation

Geon Lee
KAIST
Seoul, Republic of Korea
geonlee0325@kaist.ac.kr

Kyungho Kim
KAIST
Seoul, Republic of Korea
kkyungho@kaist.ac.kr

Kijung Shin
KAIST
Seoul, Republic of Korea
kijungs@kaist.ac.kr

Abstract

Item popularity in real-world data follows a long-tail distribution, where a few items attract most of the attention, while the majority receive much less. This disparity results in high-quality embeddings for popular (head) items, but lower-quality embeddings for unpopular (tail) items, leading to less accurate recommendations for the latter. Our observations confirm that embeddings of tail items often exhibit (1) magnitudes (i.e., norms) that are less reflective of actual popularity and (2) directions that are less effective in capturing user preferences, compared to those of head items.

To address this issue, we propose EDGE, a post-training embedding enhancement method for long-tail recommendations. EDGE employs two key strategies: (1) refining embedding magnitudes to better reflect item popularity and (2) adjusting embedding directions by leveraging knowledge from head items. Importantly, EDGE is model-agnostic and can be applied to embeddings learned from any trained recommender system. Experimental results show that EDGE significantly improves tail item recommendation performance and overall system performance, achieving up to an improvement of 211.23% in NDCG@20 over the state-of-the-art method. Our code and datasets are available at https://github.com/geon0325/EDGE.

CCS Concepts

• **Information systems → Recommender systems**.

Keywords

Recommender Systems, Long-Tail Recommendation, Post-Training

ACM Reference Format:
Geon Lee, Kyungho Kim, and Kijung Shin. 2024. Post-Training Embedding Enhancement for Long-Tail Recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3627673.3679978

1 Introduction

In various domains, item popularity follows a long-tail distribution, where a small fraction of items (head items) attract most of the attention, while the majority (tail items) receive significantly less [16]. Typically, 20% of items account for 80% of user interactions in real-world datasets [1, 10]. Consequently, recommender systems achieve higher accuracy for head items due to the abundance of available training data, but the accuracy for tail items tends to be significantly lower due to the limited training data. This imbalance negatively impacts the overall recommendation performance.

CIKM '24, October 21–25, 2024, Boise, ID, USA

ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679978

In this paper, we investigate how the amount of data available for each item influences the quality of embeddings in recommender systems. Specifically, we analyze item embeddings, with a focus on their magnitudes (i.e., norms) and directions. Magnitudes reflect the general popularity of an item, while directions capture information related to user preferences. We observe that, while embedding magnitudes effectively reflect the popularity of head items, they are less effective in capturing the popularity of tail items. Additionally, the embedding directions of tail items fail to capture user preferences as effectively as those of head items. We hypothesize that this discrepancy is due to the data scarcity of tail items.

Based on these observations, we propose EDGE (**E**mbe**d**din**g** **E**nhancement for Long-Tail Recommendation), a simple yet effective method for enhancing item embeddings. EDGE employs two key strategies: (1) refining embedding magnitudes to better reflect the actual item popularity and (2) adjusting embedding directions by leveraging reliable knowledge from head items. Importantly, EDGE operates as a *post-training* process and is *model-agnostic*. It can be applied to embeddings generated by any recommender system once the primary training process is completed. Our experimental results demonstrate that EDGE significantly improves recommendation accuracy for tail items and overall performance.

Our contributions are summarized as follows:

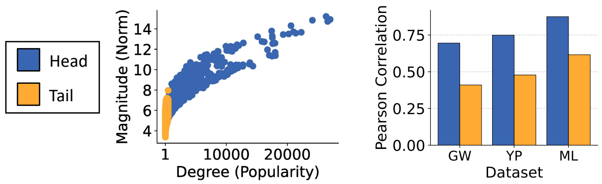
- **Observations.** We examine the quality of item embeddings, especially their magnitudes and directions, and observe that limited available data leads to low embedding quality.
- **Method.** We develop EDGE, a post-training item embedding enhancement method for long-tail recommendations. EDGE jointly refines the magnitudes and directions of item embeddings.
- **Experiments.** We perform comprehensive experiments and demonstrate that EDGE yields improvements of up to 211.23% in terms of NDCG@20 over the state-of-the-art method.

2 Preliminaries

In this section, we review some preliminaries.

Problem Definition. Consider a set of users $\mathcal{U} = \{u_1, \cdots, u_{|\mathcal{U}|}\}$ and a set of items $\mathcal{I} = \{i_1, \cdots, i_{|\mathcal{I}|}\}$. Their interactions are represented as a binary matrix $\mathbf{R} \in \{0, 1\}^{|\mathcal{U}| \times |\mathcal{I}|}$, where each entry r_{ui} is 1 if user u interacted with item i, and 0 otherwise. Our goal is to predict the relevance score $\hat{r}_{ui}$ for each unobserved user-item pair (u, i) and recommend the highest-scoring items to each user u.

Embedding-Based Recommendation. A common approach for recommendation is to learn embeddings for both users and items in a shared latent space [3, 4, 6]. The estimated relevance score $\hat{r}_{ui}$



(a) Degs. and emb. magnitudes of items in ML **(b) Correlations btw. degs. and emb. magnitudes**

Figure 1: The magnitudes of embeddings for tail items are less correlated with their popularity than those for head items, indicating that the popularity of tail items is less reflected in the magnitudes of the embeddings. More results are in [8].

for user u and item i is typically computed as the inner product of their respective embeddings $\mathbf{e}_u \in \mathbb{R}^d$ and $\mathbf{e}_i \in \mathbb{R}^d$, i.e., $\hat{r}_{ui} = \mathbf{e}_u^T \mathbf{e}_i$.
Head and Tail Items. Based on the Pareto principle [1, 10], we define *head items* $\mathcal{I}^{\mathrm{H}} \subset \mathcal{I}$ as the top 20% most frequent items and *tail items* $\mathcal{I}^{\mathrm{T}} \subset \mathcal{I}$ as the remaining 80%.

3 Analyses & Observations

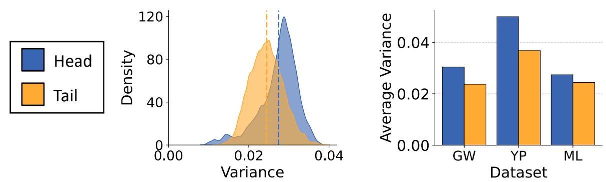
We analyze the magnitudes and directions of item embeddings and examine their roles in recommendations. Then, we investigate distinct properties of embeddings for head and tail items learned by existing recommender systems. We utilized embeddings learned from LightGCN [3] as an example; results using embeddings from other models (e.g., BPRMF [12] and NGCF [13]) can be found in [8].
Embedding Representation and Roles. The embedding $\mathbf{e}_i \in \mathbb{R}^d$ of an item i can be decomposed into its magnitude and direction as:

$$\mathbf{e}_i = \|\mathbf{e}_i\| \cdot \frac{\mathbf{e}_i}{\|\mathbf{e}_i\|},$$

where $\|\mathbf{e}_i\|$ is the *magnitude* (i.e., norm), and $\mathbf{e}_i/\|\mathbf{e}_i\|$ is the *direction* of the embedding $\mathbf{e}_i$. The relevance score $\hat{r}_{ui}$ between a user u and an item i can be expressed as:

$$\hat{r}_{ui} = \mathbf{e}_u^T \mathbf{e}_i = \|\mathbf{e}_u\|\|\mathbf{e}_i\| \left(\frac{\mathbf{e}_u^T}{\|\mathbf{e}_u\|} \frac{\mathbf{e}_i}{\|\mathbf{e}_i\|} \right) = \|\mathbf{e}_u\|\|\mathbf{e}_i\| \cos\left(\theta_{ui}\right),$$

where θ_{ui} is the angle between $\mathbf{e}_u$ and $\mathbf{e}_i$. Note that the magnitude $\|\mathbf{e}_u\|$ of the user embedding $\mathbf{e}_u$ does not affect the ranking of items for the user u. The magnitude $\|\mathbf{e}_i\|$ of the item embedding $\mathbf{e}_i$ impacts the general popularity of the item i rather than specific user preferences. The direction $\mathbf{e}_i/\|\mathbf{e}_i\|$ of the item embedding $\mathbf{e}_i$, which determines the angle θ_{ui} with the user embedding $\mathbf{e}_u$, captures the alignment with user preferences. Importantly, the magnitude and direction *jointly* determine the relevance of an item to a user.
Magnitudes of Tail Item Embeddings. We investigate the relationship between the magnitudes of item embeddings and item popularity. Our findings in Figure 1 reveal that tail items exhibit a weaker correlation between embedding magnitudes and popularity, compared to head items. This suggests that the magnitudes of embeddings for tail items are less effective at reflecting their popularity than those for head items. We conjecture that this discrepancy primarily stems from the limited data available for tail items, which challenges recommender systems to capture their popularity in the embedding magnitudes during training.



(a) Distributions of variance of angles in ML **(b) Average variance of angles in GW, YP, and ML**

Figure 2: The variance of the angles between item and user embeddings tends to be smaller for tail items than for head items, indicating that tail item embeddings are less effective at capturing user preferences. More results are in [8].

Directions of Tail Item Embeddings. We examine the relationship between the directions of item and user embeddings. Directions of properly learned item embeddings are expected to closely align with the embeddings of users who are likely to interact with the item and diverge from those of other users. To quantify this alignment, we compute the variance of the cosine similarity between each item embedding and user embeddings, i.e., the variance of $\cos(\theta_{1i}), \cdots, \cos(\theta_{|\mathcal{U}|i})$. Intuitively, a higher variance indicates that the item embedding has greater discriminative power in identifying users who are close to the item and those who are not. As shown in Figure 2, our observations reveal that tail items tend to have smaller variances in the cosine similarities with user embeddings, suggesting that tail items are less effective at capturing user preferences. We conjecture that this issue is also primarily rooted in the data scarcity associated with tail items.

4 Proposed Method

We present EDGE, a simple yet effective post-training embedding enhancement method. EDGE jointly refines the magnitude and direction of item embeddings. It is *model-agnostic*, i.e., applicable to embeddings learned from any recommendation model.
Magnitude Refinement. To better reflect the popularity of items within the magnitude of their embeddings, we refine the magnitude $\|\mathbf{e}_i\|$ of the embedding $\mathbf{e}_i$ of item i as follows:

$$\|\mathbf{e}_i\| \rightarrow \|\mathbf{e}_i\|^{1-\alpha} \cdot d_i^{\beta},$$

where $\alpha \in [0,1]$ and $\beta \in [0,1]$ are hyperparameters, and d_i is the degree of item i in the training set. Here, increasing α reduces the impact on the magnitudes induced by the model, and increasing β adjusts the magnitude to better reflect item popularity.
Direction Refinement. We refine the direction of item embeddings by leveraging knowledge from head items, which have been trained on abundant data and thus are effective at capturing user preferences, as empirically verified in Section 3. For the embedding $\mathbf{e}_i$ of item i, we compute its *head-inferred* embedding as follows:

$$\mathbf{e}_i^{\mathrm{H}} = \sum_{j \in \mathcal{I}^{\mathrm{H}}} \gamma_{ij} \frac{\mathbf{e}_j}{\|\mathbf{e}_j\|}, \quad \text{where} \quad \gamma_{ij} = \frac{\exp\left(\cos\left(\theta_{ij}\right)/\tau\right)}{\sum_{k \in \mathcal{I}^{\mathrm{H}}} \exp\left(\cos\left(\theta_{ik}\right)/\tau\right)}, \quad (1)$$

and $\tau \in (0, \infty)$ is a hyperparameter. The directions of head item embeddings are aggregated based on their relevance to item i (specifically, by the cosine similarity with the embedding of item i). Then,

Table 1: EDGE significantly enhances the recommendation performance of BPRMF, NGCF, and LightGCN, in both unbiased and biased test settings across all datasets. It consistently outperforms TTEN [5], the state-of-the-art post-training embedding enhancement method for long-tail recommendations. The improvements are measured relative to TTEN's performance.

		Unbiased Setting						Biased Setting					
		Gowalla		Yelp		MovieLens		Gowalla		Yelp		MovieLens	
		Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
BPRMF	Original	0.1367	0.0869	0.0471	0.0332	0.0610	0.0344	0.1532	0.0899	0.0682	0.0366	0.1126	0.0481
	+ TTEN	0.1445	0.0926	0.0528	0.0363	0.0647	0.0358	0.1532	0.0899	0.0682	0.0366	0.1126	0.0481
	+ EDGE	**0.1601**	**0.0992**	**0.0716**	**0.0448**	**0.0776**	**0.0428**	**0.2160**	**0.1412**	**0.1089**	**0.0623**	**0.2929**	**0.1497**
Improvement		10.80%	7.13%	35.61%	23.42%	19.94%	19.55%	40.99%	57.06%	59.68%	70.22%	160.12%	211.23%
NGCF	Original	0.0897	0.0560	0.0272	0.0211	0.0009	0.0011	0.1893	0.1190	0.0937	0.0523	0.1704	0.0857
	+ TTEN	0.1449	0.0939	0.0574	0.0379	0.0540	0.0302	0.1899	0.1195	0.0937	0.0523	0.2998	0.1469
	+ EDGE	**0.1602**	**0.0991**	**0.0643**	**0.0412**	**0.0595**	**0.0314**	**0.2058**	**0.1339**	**0.1097**	**0.0633**	**0.3285**	**0.1706**
Improvement		10.56%	5.54%	12.02%	8.71%	10.19%	3.97%	8.37%	12.05%	17.08%	21.03%	9.57%	16.13%
LightGCN	Original	0.0751	0.0449	0.0173	0.0157	0.0106	0.0067	0.2054	0.1322	0.1103	0.0617	0.2715	0.1271
	+ TTEN	0.1408	0.0915	0.0549	0.0368	0.0658	0.0355	0.2079	0.1348	0.1103	0.0617	0.2768	0.1305
	+ EDGE	**0.1569**	**0.0971**	**0.0647**	**0.0413**	**0.0769**	**0.0410**	**0.2173**	**0.1421**	**0.1155**	**0.0656**	**0.3211**	**0.1634**
Improvement		11.43%	6.12%	17.85%	12.23%	16.87%	15.49%	4.52%	5.42%	4.71%	6.32%	16.00%	25.21%

the direction of the embedding $\mathbf{e}_i$ is refined as:

$$\frac{\mathbf{e}_i}{\|\mathbf{e}_i\|} \to \frac{\tilde{\mathbf{e}}_i}{\|\tilde{\mathbf{e}}_i\|}, \quad \text{where} \quad \tilde{\mathbf{e}}_i = \frac{\mathbf{e}_i}{\|\mathbf{e}_i\|} + \lambda \frac{\mathbf{e}_i^{\mathrm{H}}}{\|\mathbf{e}_i^{\mathrm{H}}\|},$$

and $\lambda \in [0, \infty)$ is a hyperparameter. When $\lambda > 0$, the direction of the head-inferred embedding $\mathbf{e}_i^{\mathrm{H}}$ is used to refine the direction of the embedding $\mathbf{e}_i$. This enrichment with reliable information from head items, inferred from many interactions, potentially results in embeddings with clearer information about user preferences.

Joint Refinement. We jointly apply both magnitude and direction refinement schemes to enhance the item embeddings as follows:

$$\mathbf{e}_i \to \|\mathbf{e}_i\|^{1-\alpha} \cdot d_i^{\beta} \cdot \frac{\tilde{\mathbf{e}}_i}{\|\tilde{\mathbf{e}}_i\|}, \quad \text{where} \quad \tilde{\mathbf{e}}_i = \frac{\mathbf{e}_i}{\|\mathbf{e}_i\|} + \lambda \frac{\mathbf{e}_i^{\mathrm{H}}}{\|\mathbf{e}_i^{\mathrm{H}}\|},$$

and the head-inferred embedding $\mathbf{e}_i^{\mathrm{H}}$ is defined in Eq. (1). Note that Kim et al. [5] introduce α to refine embedding magnitudes, and EDGE enhances this approach by introducing β to account for item popularity and λ to leverage knowledge from head items. This joint refinement in magnitude and direction significantly improves recommendation performance, as demonstrated in Section 5.

5 Experiments

We conduct comprehensive experiments on EDGE to evaluate its accuracy, effectiveness, and efficiency.

5.1 Experimental Settings

We discuss the experimental settings. For full details, refer to [8].

Datasets. We used three publicly available datasets that are widely used in recommender systems research: Gowalla (GW) [2], Yelp (YP),[1] and MovieLens (ML).[2] These datasets were processed following [5]. Some basic statistics of each dataset are given in [8].

Evaluation. The performances of all models are evaluated using Recall@20 and NDCG@20 metrics, and the training, validation, and test sets are split with a ratio 7:1:2, following [5]. We employ two distinct test sets for a comprehensive evaluation. An **unbiased** test set, used in [5, 11, 14, 17], is constructed so that each item appears an equal number of times in the test interactions. A **biased** test set is constructed by uniform random sampling of interactions, leading to a similar long-tail distribution as the training set. While EDGE is applicable to any embedding-based recommendation model, we use LightGCN [3] as the default base model unless otherwise stated.

Implementation. Our implementation of EDGE is based on the framework provided by [5]. For the hyperparameters, α, β, τ, and λ, we explored the following ranges: $\alpha \in \{0.0, 0.2, \ldots, 1.0\}$, $\beta \in \{0.0, 0.1, 0.2\}$, $\tau \in \{0.01, 0.05, 0.2\}$, and $\lambda \in \{0.0, 0.2, \ldots, 2.0\}$. Inspired by the empirical effectiveness demonstrated in [5], we used the SSM (Sampled Softmax) loss [15] to train the models, which generally yielded better results than the BPR loss [12] (refer to [8]).

[1] https://www.yelp.com/dataset

[2] https://grouplens.org/datasets/movielens

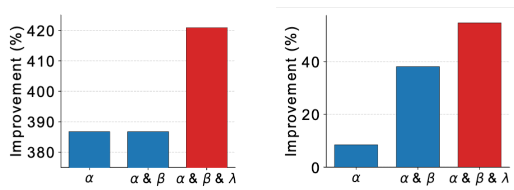
Table 2: EDGE outperforms its competitors in long-tail recommendations. Results marked * are from [5].

	Gowalla		Yelp		MovieLens	
	Recall	NDCG	Recall	NDCG	Recall	NDCG
BPRMF	0.137	0.087	0.047	0.033	0.061	0.034
LightGCN	0.075	0.045	0.017	0.016	0.011	0.007
IPS *	0.057	0.035	0.012	0.012	0.018	0.010
GRAD *	0.081	0.053	-	-	-	-
MACR *	0.100	0.054	0.047	0.027	0.057	0.028
BiGNN *	0.108	0.059	0.046	0.027	0.059	0.032
TTEN	0.141	0.092	0.055	0.037	0.066	0.036
EDGE	**0.157**	**0.097**	**0.065**	**0.041**	**0.077**	**0.041**

5.2 Experimental Results

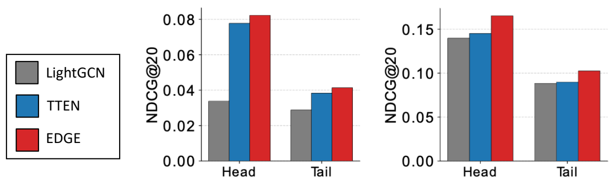
We discuss our experimental results discussed as follows.

Comparison with a Post-Training Method. We evaluate EDGE in Table 1 under both unbiased and biased test settings. Since EDGE is model-agnostic, it can be applied to any embedding-based recommender system. When applied to three models, BPRMF [12], NGCF [13], and LightGCN [3], EDGE consistently and significantly improves recommendation performance of the base models. Additionally, EDGE outperforms TTEN [5], the state-of-the-art post-training embedding enhancement method for long-tail recommendations. While TTEN shows limited effectiveness in the biased setting, EDGE is effective in both unbiased and biased settings. These results demonstrate the effectiveness of EDGE's strategy for refining both the magnitude and direction of embeddings.



(a) Unbiased Setting (b) Biased Setting

Figure 3: Incorporating α and β for magnitude refinement and λ for direction refinement incrementally increases the average NDCG@20 improvements of BPRMF, NGCF, and LightGCN in both unbiased and biased test settings.



(a) Unbiased Setting (b) Biased Setting

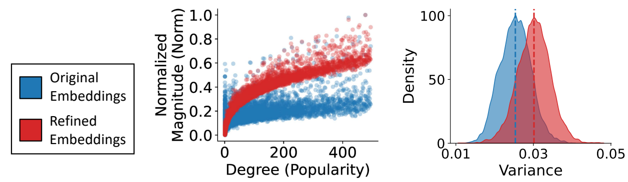
Figure 4: EDGE improves the performance for both head and tail items by LightGCN [3] and outperforms TTEN [5] in MovieLens. More results can be found in [8].

Comparison with Training-Based Methods. We compare EDGE with *training-based* methods, including IPS [9], GRAD [11], MACR [14], and BiGNN [7], which *learn* embeddings via training (e.g., gradient descent). These models are specifically designed to address long-tail distributions, and we follow their common unbiased test settings for evaluation. As shown in Table 2, EDGE outperforms all competitors across all datasets, demonstrating its effectiveness in enhancing embeddings as a post-training process.

Ablation Study. To examine the effects of each component of EDGE, we compare the performance of its variants in Figure 3. Introducing α and λ is effective in both unbiased and biased settings. Introducing β is effective in the biased setting but has minimal effectiveness in the unbiased setting (i.e., it was tuned closely to 0), where the training and test popularities significantly differ. EDGE, incorporating α, β, and λ, achieves the best performance in both unbiased and biased test settings, with an average improvement of 420.88% and 54.71%, respectively, over the original models.

Performance on Head & Tail Items. We evaluate the respective performance of head and tail items. As shown in Figure 4, EDGE improves LightGCN's performance for both head and tail items in unbiased and biased settings. In the unbiased setting, where item popularities significantly differ in training and test sets, the improvement for both head and tail items is notable as they easily benefit from the adjustment of magnitude, which reflects popularity, through α (recall that β has minimal effect in the setting). Tail items, which are inherently difficult to recommend accurately due to limited data, also show non-trivial improvements in both settings.

Enhancement of Tail Item Embeddings. We examine how the magnitudes and directions of tail item embeddings are enhanced after applying EDGE. As shown in Figure 5, EDGE refines both the magnitude and direction of tail item embeddings. Specifically, the correlation between embedding magnitude and popularity increases, as does the average variance of similarity with user embeddings (see Section 2). These enhancements collectively lead to more accurate recommendations of tail items, as shown in Figure 4.



(a) Magnitude Refinement (b) Direction Refinement

Figure 5: EDGE enhances tail item embeddings. Compared to the original embeddings, the refined embeddings exhibit magnitudes that better align with item popularity and directions that are more discriminative towards user preferences.

Speed. Since EDGE is applied as a post-training step, it does not require graph convolution or gradient descent. Consequently, EDGE requires minimal execution time compared to the training time needed to learn the embeddings. For instance, LightGCN (with the SSM loss) requires an average of 11.30, 22.11, and 167.25 seconds per epoch for Gowalla, Yelp, and MovieLens, respectively. In contrast, EDGE takes 0.001, 0.011, and 0.013 seconds to adjust the embeddings for the same datasets, demonstrating the efficiency of EDGE.[3]

6 Related Work

Long-tail data arise in various domains, and recommendations on such data are crucial in practical applications [16]. To better recommend both tail and head items, most prior research has developed specific strategies during the training process. These include techniques for item re-weighting [9], gradient adjustment [11], specialized loss functions [15], and embedding disentanglement [17]. Recently, TTEN [5], which adjusts the magnitude of item embeddings as a post-training process, has shown state-of-the-art performance. However, TTEN overlooks the importance of the quality of embedding directions. In addition, its effectiveness diminishes when evaluated in biased settings, as shown in Section 5. EDGE builds upon the post-training approach and enhances both the magnitudes and directions of embeddings. We empirically demonstrate that EDGE is effective in both unbiased and biased test settings.

7 Conclusions

In this paper, we present EDGE, a simple yet effective embedding enhancement method for long-tail recommendations. Motivated by our empirical analyses regarding the quality of item embeddings, EDGE jointly refines the magnitude and direction of item embeddings. Importantly, EDGE operates as a post-training process and is model-agnostic, making it applicable to embeddings learned by any recommender system. Our experimental results demonstrate that EDGE significantly improves the recommendation performance of various models for both tail and head items.

Acknowledgements. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00406985, 90%). This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2019-II190075, Artificial Intelligence Graduate School Program (KAIST), 10%).

[3]The experiments were conducted using a machine with an RTX 8000 GPU.

References

[1] George EP Box and R Daniel Meyer. 1986. An analysis for unreplicated fractional factorials. *Technometrics* (1986), 11–18.

[2] Eunjoon Cho, Seth A Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *KDD*.

[3] 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*.

[4] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*.

[5] Dain Kim, Jinhyeok Park, and Dongwoo Kim. 2023. Test-Time Embedding Normalization for Popularity Bias Mitigation. In *CIKM*.

[6] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[7] Yingshuai Kou, Neng Gao, Yifei Zhang, Chenyang Tu, and Cunqing Ma. 2022. BiGNN: A Bilateral-Branch Graph Neural Network to Solve Popularity Bias in Recommendation. In *ICTAI*.

[8] Geon Lee, Kyungho Kim, and Kijung Shin. 2024. Post-Training Embedding Enhancement for Long-Tail Recommendation (Supplementary Document). https://github.com/geon0325/EDGE/blob/main/supplementary.pdf.

[9] Dawen Liang, Laurent Charlin, and David M Blei. 2016. Causal inference for recommendation. In *Causation: Foundation to Application, Workshop at UAI*.

[10] Mark EJ Newman. 2005. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics* 46, 5 (2005), 323–351.

[11] Weijieying Ren, Lei Wang, Kunpeng Liu, Ruocheng Guo, Lim Ee Peng, and Yanjie Fu. 2022. Mitigating popularity bias in recommendation with unbalanced interactions: A gradient perspective. In *ICDM*.

[12] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*.

[13] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *SIGIR*.

[14] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *KDD*.

[15] Jiancan Wu, Xiang Wang, Xingyu Gao, Jiawei Chen, Hongcheng Fu, and Tianyu Qiu. 2024. On the effectiveness of sampled softmax loss for item recommendation. *ACM Transactions on Information Systems* 42, 4 (2024), 1–26.

[16] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. 2023. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 9 (2023), 10795–10816.

[17] Yu Zheng, Chen Gao, Xiang Li, Xiangnan He, Yong Li, and Depeng Jin. 2021. Disentangling user interest and conformity for recommendation with causal embedding. In *WWW*.