Inductive Transfer Learning for Graph-Based Recommenders

Florian Grötschla* **Elia Trachsel*** **Luca A. Lanzendörfer** **Roger Wattenhofer**

ETH Zurich
{fgroetschla, trachsele, lanzendoerfer, wattenhofer}@ethz.ch

Abstract

Graph-based recommender systems are commonly trained in transductive settings, which limits their applicability to new users, items, or datasets. We propose NBF-Rec, a graph-based recommendation model that supports inductive transfer learning across datasets with disjoint user and item sets. Unlike conventional embedding-based methods that require retraining for each domain, NBF-Rec computes node embeddings dynamically at inference time. We evaluate the method on seven real-world datasets spanning movies, music, e-commerce, and location check-ins. NBF-Rec achieves competitive performance in zero-shot settings, where no target domain data is used for training, and demonstrates further improvements through lightweight fine-tuning. These results show that inductive transfer is feasible in graph-based recommendation and that interaction-level message passing supports generalization across datasets without requiring aligned users or items.

1 Introduction

Recommender systems play a central role in many digital platforms, where they personalize content by modeling interactions between users and items. Graph-based approaches have proven effective in this setting, as they capture complex relational dependencies through message passing on user-item interaction graphs. However, most graph-based recommenders are trained and evaluated within a single dataset, making strong assumptions about the availability of domain-specific interaction histories. This limits their ability to generalize to new users, items, or domains, and restricts their applicability in real-world scenarios where data is sparse or fragmented across platforms. In contrast, transfer learning has enabled substantial progress in fields such as natural language processing and computer vision, where models pretrained on large datasets can be adapted to new tasks with limited supervision. Despite this success, transfer learning remains underexplored in the context of graph-based recommender systems. Existing methods often rely on transductive training paradigms or assume overlapping user or item sets across domains. These constraints hinder their ability to perform recommendation in settings where no such overlap exists and where target domain data is unavailable during training. We address this gap by introducing NBF-Rec, a graph-based recommendation model that supports inductive transfer learning across disjoint user-item graphs. NBF-Rec computes node representations dynamically via message passing, without relying on pretrained embeddings or access to the target domain during training. This design enables zero-shot recommendation in new domains and supports further adaptation through fine-tuning when limited target interactions are available. We evaluate NBF-Rec on seven real-world datasets spanning diverse application areas, including movies, music, e-commerce, and location-based services. Our experiments show that the model achieves competitive performance in zero-shot scenarios, and that fine-tuning on a small subset of

*Equal contribution.

New Perspectives in Graph Machine Learning Workshop at NeurIPS 2025

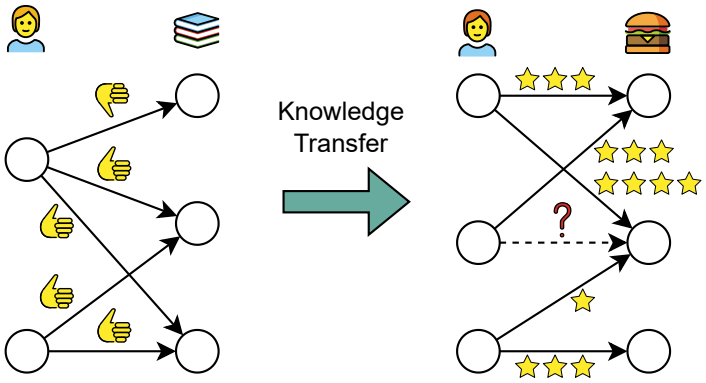

Figure 1: Schematic overview of NBF-Rec's inductive transfer learning pipeline. The model is trained on user-item interaction graphs from one or more source domains and applied to unseen target domains. Transferable patterns are learned from structural and interaction features without requiring shared users or items.

target interactions leads to further improvements. These results demonstrate the viability of inductive transfer learning in graph-based recommendation.

2 Related Work

Graph Neural Networks (GNNs) have become central in recommendation systems due to their ability to model complex user-item relationships. Early models like LightGCN [1] achieved strong performance by simplifying message passing, but were transductive in nature and failed to generalize to unseen nodes or graphs. To enable inductive reasoning, models such as MINERVA [2], GraIL [3], and RED-GNN [4] introduced path- and subgraph-based methods. NBFNet [5], which forms the foundation of our approach, learns to aggregate path-based messages in a query-dependent manner, supporting inductive link prediction. More recent work has explored transfer learning in GNNs to improve generalization across domains. ULTRA [6] extended NBFNet for zero-shot knowledge graph completion by jointly reasoning over relation and entity graphs. In recommendation, most cross-domain models assume shared users or items [7], limiting their applicability. Other works employ adversarial training [8], contrastive disentanglement [9–12], or meta-learning [13] to address cross-domain generalization, but often rely on aligned entity spaces or domain-specific supervision. Efforts leveraging pretrained language or vision models, such as U-BERT [14] and ChatRec [15], have shown promise for cross-domain recommendation but typically require rich side information. In contrast, NBF-Rec operates solely on interaction graphs, without depending on textual or visual content. Contrastive learning methods like SimGCL [16] and NESCL [17] enhance transferability by regularizing representations, though they focus primarily on improving performance within a single domain. Large-scale pretrained models have recently been explored as a unified solution to recommendation tasks. P5 [18] formulates personalized recommendation as a text-to-text problem and leverages large language models to handle multiple tasks across domains in a unified framework. Similarly, GPTRec [19] adapts generative pretrained transformers to model user behavior sequences for next-item prediction. While these models achieve strong performance in zero-shot and multitask settings, they require substantial pretraining and inference resources. In contrast, NBF-Rec provides a lightweight alternative based on graph message passing, enabling inductive transfer across datasets with disjoint user and item sets. To our knowledge, no prior graph-based recommender has demonstrated scalable, inductive transfer across disjoint user-item graphs in a purely interaction-driven setting.

3 Methodology

3.1 Problem Formulation

Recommender systems can be formulated as a *link prediction* task on a bipartite user-item graph. Let $\mathcal{U}$ be the set of users and $\mathcal{I}$ be the set of items. The history of interaction between users and items is represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{U} \times \mathcal{I}$ denotes the interactions observed between users and items. Each edge $(u, i) \in \mathcal{E}$ represents an interaction (e.g., user buying a product, watching a movie, or listening to a song). Given an incomplete graph $\mathcal{G}$, our goal is to predict missing edges, i.e., to recommend new items to users.

3.2 NBF-Rec Model Architecture

NBF-Rec is an inductive link prediction model that builds on neural Bellman-Ford networks (NBFNet) and integrates edge features into the message-passing process. Edge features capture rich interaction-level context and enable the model to distinguish different types and strengths of user-item interactions. For example, in the Epinions dataset, edge features include category, rating, helpfulness, and timestamp, which provide nuanced signals about user preferences and item relevance. Similarly, the LastFM dataset includes listening counts as edge features, reflecting users' interaction frequency and intensity with specific artists. Unlike traditional embedding-based approaches, which require retraining when new users or items appear, NBF-Rec dynamically computes node embeddings for previously unseen graphs. Given a user u, the message-passing procedure is defined as follows:

Initialization. Each node v is initialized according to the standard NBFNet scheme: $h_v^{(0)} = \mathbf{1}(u = v)$, which ensures that all information propagates from the query user u. Additionally, we define g as the function that generates edge embeddings from raw edge features r:

$$g(r) = \text{MLP}_{\text{emb}}\Big(\text{MLP}_{\text{proj}}(r)\Big). \tag{1}$$

This function consists of a dataset-specific projection MLP followed by a backbone embedding MLP.

Message Computation. At each layer t, the messages for a node v are computed as:

$$M_v^{(t)} = \Big\{\text{MESSAGE}\Big(h_x^{(t-1)}, \mathbf{w}_q(x, r, v)\Big) \mid (x, r, v) \in \mathcal{E}(v)\Big\}, \tag{2}$$

where the edge weight is given by:

$$\mathbf{w}_q(x, r, v) = \text{MLP}_t\big(g(r)\big). \tag{3}$$

Here, edge features r are transformed into layer-specific embeddings via MLP_t. The message function is implemented using a non-parametric *DistMult* operation [20].

Node Update. Each node v updates its representation by aggregating incoming messages along with its initial embedding:

$$h_v^{(t)} = \text{UPDATE}\Big(h_v^{(t-1)},\ \text{AGGREGATE}\Big(M_v^{(t)} \cup \{h_v^{(0)}\}\Big)\Big). \tag{4}$$

The aggregation function is a summation operation, followed by a learnable update function comprising a linear transformation, layer normalization, and an activation function.

Score Generation. After T layers of message passing, we compute logit scores for each node v using a final MLP that projects the concatenation of the last-layer embedding and the initial embedding to a scalar value:

$$\text{score}(u, q, v) = \text{MLP}_{\text{score}}\Big(\text{concat}\big(h_v^{(T)}, h_v^{(0)}\big)\Big). \tag{5}$$

Notably, our model does not learn node-specific parameters, enabling node-inductive generalization. Unlike traditional embedding-based methods such as LightGCN [1], which precompute node embeddings and apply simple similarity functions for recommendation, our approach dynamically performs message passing for each query.

Table 1: Dataset statistics for the seven real-world benchmarks used in our evaluation. For each dataset, we report the number of users, items, and observed interactions, as well as the primary evaluation metric used in our benchmarking.

Dataset	**#Users**	**#Items**	**#Interactions**	**Metric**
ML-1M [22]	5,950	2,811	364,654	Hits@10
LastFM [23]	1,867	1,867	39,717	Hits@10
Amazon B. [24]	52,204	57,289	293,912	Hits@10
Gowalla [25]	29,858	70,839	712,504	NDCG@20
Epinions [26]	21,008	13,887	266,791	Hits@10
BookX [27]	12,720	18,318	276,334	Hits@10
Yelp18 [28]	31,668	38,048	1,097,007	NDCG@20

3.3 Training

Following the approach by Zhu et al. [21], we train the model by minimizing a cross-entropy loss over positive and negative triplets:

$$\mathcal{L} = -\log p(u, q, v) - \sum_{i=1}^{n} \frac{1}{n} \log(1 - p(u'_i, q, v'_i)), \tag{6}$$

where (u, q, v) is a positive interaction, and $\{(u'_i, q, v'_i)\}_{i=1}^{n}$ are negative samples generated by corrupting either the head u or the tail v. The negative samples are strict negatives, meaning they do not appear in the training dataset and are selected uniformly at random. This loss encourages the model to assign higher probabilities to positive interactions while minimizing the likelihood of negative ones. To optimize our model, we remove easy edges (i.e. batch edges) from the convolution graph during training. The training dataset defines the underlying graph for convolution, meaning that for every batch edge, the corresponding edge is already present in the convolution graph. If left unchanged, this setup would allow the model to trivially learn direct interactions rather than extracting meaningful relational patterns. By removing these edges, we force the model to rely on non-trivial paths.

3.4 Computational Complexity

NBF-Rec performs inference efficiently through dynamic message passing without precomputing user and item embeddings. A forward pass consists of (1) edge embedding transformation ($\mathcal{O}(|E|)$), (2) message passing ($\mathcal{O}(T|E|)$, with T layers), and (3) score computation ($\mathcal{O}(|V|)$), resulting in overall linear complexity in the number of nodes and edges. While NBF-Rec incurs higher inference costs than models such as LightGCN, which precompute node embeddings, it enables transfer to unseen nodes, a capability absent in transductive architectures.

4 Experimental Evaluation

4.1 Datasets

We evaluate NBF-Rec on seven real-world recommendation datasets spanning movies (MovieLens-1M), music (LastFM), e-commerce (Amazon Beauty), location check-ins (Gowalla), product reviews (Epinions), and local business ratings (Yelp18). Each dataset is modeled as a bipartite user-item graph, with edge features such as ratings, timestamps, interaction counts, or metadata incorporated when available. MovieLens-1M contains around 1M ratings from 5.9K users on 2.8K movies. LastFM includes implicit feedback (e.g., play counts) between 1.8K users and artists. Amazon Beauty and BookX represent large-scale product domains, with over 250K interactions each. Gowalla captures spatio-temporal check-ins, while Epinions includes explicit ratings and user trust relationships. Yelp18 contains over 1M user-business reviews with textual and temporal metadata. Dataset statistics are summarized in Table 1.

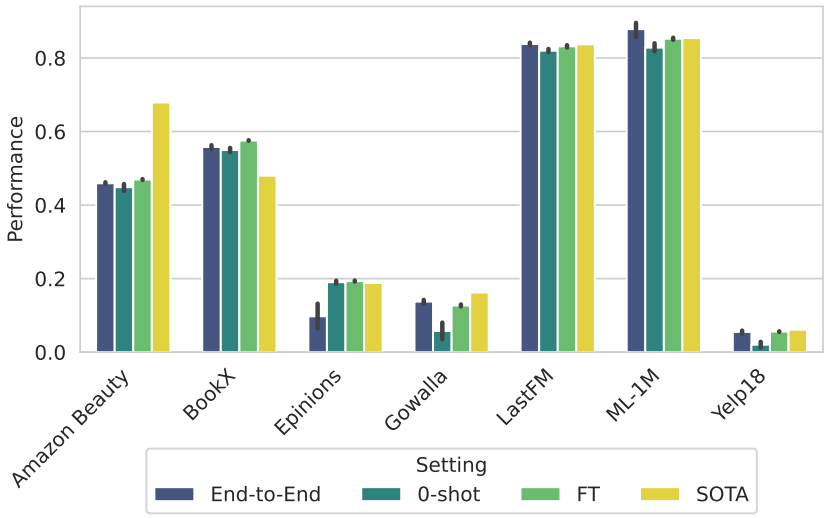

Figure 2: Performance of NBF-Rec in zero-shot, fine-tuned, and end-to-end settings across seven datasets. Metrics are Hits@10 or NDCG@20, depending on the dataset (see Table 1). Bars indicate 95% confidence intervals over 3 runs. Zero-shot transfer achieves competitive results.

4.2 Experimental Setup

For each dataset, we evaluate NBF-Rec in three settings:

- **End-to-End Training**: The model is trained and tested on the same dataset.
- **Zero-Shot**: NBF-Rec is pretrained on one dataset and directly tested on another without fine-tuning. Embedding MLPs are randomly initialized.
- **Fine-Tuning**: The whole pretrained model is fine-tuned on the target dataset.

We pretrain our models on datasets using both single-graph and multi-graph configurations, where we train on multiple datasets. Alongside our model, we also evaluate a simplified version of NBFNet that relies solely on structural graph information. For evaluation, we use **Hits@K** and **NDCG@K**. Hits@K measures whether a relevant item appears in the top K recommendations, while NDCG@K incorporates ranking information by normalizing the discounted cumulative gain (DCG) against the ideal DCG (IDCG). While Hits@K provides a binary assessment of whether a relevant item appears in the top-K recommendations, NDCG@K captures both the presence and ranking of relevant items in the list.

4.3 Results and Analysis

We evaluate the ability of NBF-Rec to generalize across recommendation domains in two inductive transfer settings: (i) zero-shot transfer, where no target domain data is seen during training, and (ii) fine-tuning, where the target data is used for supervised adaptation. Unless otherwise noted, results are derived from a model pretrained on Amazon Beauty and Epinions. We report SOTA performance for each dataset based on results from prior work (ML-1M [29], LastFM [30], Amazon Beauty [31], Gowalla [32], Epinions [28], BookX [29], Yelp18 [32]). Figure 2 shows the performance of NBF-Rec in these settings compared to a domain-specific model trained end-to-end on each dataset. We observe that zero-shot transfer yields competitive results across most datasets. For example, on MovieLens-1M, BookX, and Yelp18, the zero-shot model achieves performance within 5% of the fully supervised baseline. This demonstrates that NBF-Rec learns representations and message-passing procedures that generalize to unseen user-item graphs without requiring retraining. Fine-tuning the pretrained model on the target domain consistently improves performance across all datasets. The relative improvement is most pronounced on LastFM and Gowalla, where the zero-shot setting initially underperforms. These results suggest that even modest amounts of in-domain supervision suffice

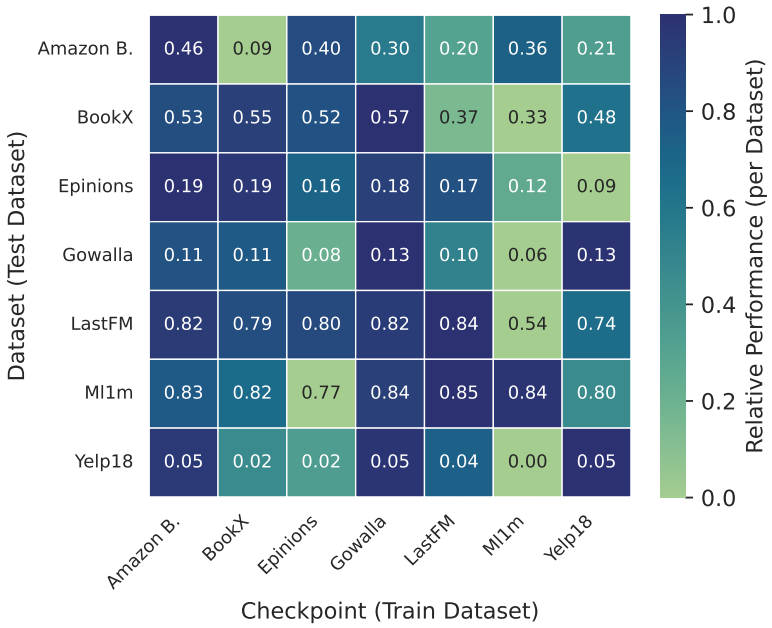

Figure 3: Cross-dataset transfer heatmap: each column indicates the training dataset and each row the test dataset. Cell colors represent relative performance (normalized by row). NBF-Rec achieves strong transfer when trained on Amazon Beauty, outperforming several dataset-specific models.

to close the gap to fully supervised models. Importantly, NBF-Rec supports this adaptation without reinitialization or retraining from scratch.

Cross-Dataset Transfer. We analyze dataset-to-dataset transferability based on the heatmap in Figure 3. Along the diagonal, models are evaluated in a zero-shot setting using only the pretrained backbone checkpoint, without end-to-end training on the target domain. Notably, the diagonal does not always yield the best performance: for instance, BookX and Amazon Fashion achieve higher accuracy when transferred from other datasets than from their own backbone. This suggests that pretraining on different domains can provide more effective inductive biases than pretraining on the target domain itself. Edge features that are less informative, as observed in BookX and LastFM, generally reduce transferability. However, rich edge features alone do not guarantee successful transfer, as illustrated by Yelp's relatively weak performance. Conversely, Gowalla, a large graph with fewer edge attributes, transfers effectively. Finally, transferability is not symmetric: for example, models trained on LastFM transfer well to MovieLens-1M, whereas the reverse transfer is less effective.

Comparison with NBFNet. We compare NBF-Rec to its base model, NBFNet, which solely relies on the graphs' structural information to assess the impact of our modifications (e.g., the inclusion of edge embeddings) on transferability and overall performance. Figure 4 (Appendix) shows that in the zero-shot and fine-tuning settings, NBFRec consistently outperforms NBFNet. Even under end-to-end training, NBF-Rec usually performs on par with or slightly better than NBFNet. These results confirm that NBF-Rec generalizes the NBFNet architecture by supporting inductive transfer with node and edge embeddings while maintaining strong in-domain performance.

5 Conclusions

We presented NBF-Rec, a graph-based recommender designed for inductive transfer learning across datasets with disjoint user and item sets. Unlike transductive models, NBF-Rec computes node representations dynamically and supports generalization without retraining. Experiments on seven real-world datasets show that NBF-Rec performs competitively in zero-shot settings and improves

further with fine-tuning, often matching end-to-end baselines. These results demonstrate the feasibility of transfer in graph-based recommendation without shared entities. Our architecture primarily serves to demonstrate that inductive transfer is a viable approach for graph-based recommendation. While results vary across domains and our evaluation is limited to medium-scale datasets, the findings indicate that effective cross-domain generalization is achievable and highlight transfer learning as a promising direction for future recommender systems.

References

[1] 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*, pages 639–648, 2020.

[2] Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. *arXiv preprint arXiv:1711.05851*, 2017.

[3] Komal Teru, Etienne Denis, and Will Hamilton. Inductive relation prediction by subgraph reasoning. In *International conference on machine learning*, pages 9448–9457. PMLR, 2020.

[4] Yongqi Zhang and Quanming Yao. Knowledge graph reasoning with relational digraph. In *Proceedings of the ACM web conference 2022*, pages 912–924, 2022.

[5] Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal Xhonneux, and Jian Tang. Neural bellman-ford networks: A general graph neural network framework for link prediction. *Advances in neural information processing systems*, 34:29476–29490, 2021.

[6] Mikhail Galkin, Xinyu Yuan, Hesham Mostafa, Jian Tang, and Zhaocheng Zhu. Towards foundation models for knowledge graph reasoning. *arXiv preprint arXiv:2310.04562*, 2023.

[7] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. Cross-domain recommendation: challenges, progress, and prospects. *arXiv preprint arXiv:2103.01696*, 2021.

[8] Chenglin Li, Mingjun Zhao, Huanming Zhang, Chenyun Yu, Lei Cheng, Guoqiang Shu, Beibei Kong, and Di Niu. Recguru: Adversarial learning of generalized user representations for cross-domain recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pages 571–581, 2022.

[9] Hourun Li, Yifan Wang, Zhiping Xiao, Jia Yang, Changling Zhou, Ming Zhang, and Wei Ju. Disco: graph-based disentangled contrastive learning for cold-start cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12049–12057, 2025.

[10] Ruohan Zhang, Tianzi Zang, Yanmin Zhu, Chunyang Wang, Ke Wang, and Jiadi Yu. Disentangled contrastive learning for cross-domain recommendation. In Xin Wang, Maria Luisa Sapino, Wook-Shin Han, Amr El Abbadi, Gill Dobbie, Zhiyong Feng, Yingxiao Shao, and Hongzhi Yin, editors, *Database Systems for Advanced Applications*, pages 163–178, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-30672-3.

[11] Jing Liu, Lele Sun, Weizhi Nie, Peiguang Jing, and Yuting Su. Graph disentangled contrastive learning with personalized transfer for cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8769–8777, 2024.

[12] Xubin Ren, Lianghao Xia, Jiashu Zhao, Dawei Yin, and Chao Huang. Disentangled contrastive collaborative filtering. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1137–1146, 2023.

[13] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. A meta-learning perspective on cold-start recommendations for items. *Advances in neural information processing systems*, 30, 2017.

[14] Zhaopeng Qiu, Xian Wu, Jingyue Gao, and Wei Fan. U-bert: Pre-training user representations for improved recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4320–4327, 2021.

[15] Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*, 2023.

[16] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[17] Peijie Sun, Le Wu, Kun Zhang, Xiangzhi Chen, and Meng Wang. Neighborhood-enhanced supervised contrastive learning for collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 36(5):2069–2081, 2023.

[18] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems*, pages 299–315, 2022.

[19] Aleksandr V Petrov and Craig Macdonald. Generative sequential recommendation with gptrec. *arXiv preprint arXiv:2306.11114*, 2023.

[20] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

[21] Zhaocheng Zhu, Chence Shi, Zuobai Zhang, Shengchao Liu, Minghao Xu, Xinyu Yuan, Yangtian Zhang, Junkun Chen, Huiyu Cai, Jiarui Lu, et al. Torchdrug: A powerful and flexible machine learning platform for drug discovery. *arXiv preprint arXiv:2202.08320*, 2022.

[22] 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.

[23] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *Proceedings of the fifth ACM conference on Recommender systems*, pages 387–388, 2011.

[24] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024.

[25] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090, 2011.

[26] Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. Trust management for the semantic web. In *International semantic Web conference*, pages 351–368. Springer, 2003.

[27] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32, 2005.

[28] 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*, pages 165–174, 2019.

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

[30] Xia Ning and George Karypis. Slim: Sparse linear methods for top-n recommender systems. In *2011 IEEE 11th international conference on data mining*, pages 497–506. IEEE, 2011.

[31] Alejo Lopez-Avila, Jinhua Du, Abbas Shimary, and Ze Li. Positional encoding is not the same as context: A study on positional encoding for sequential recommendation. *arXiv preprint arXiv:2405.10436*, 2024.

[32] Peijie Sun, Le Wu, Kun Zhang, Xiangzhi Chen, and Meng Wang. Neighborhood-enhanced supervised contrastive learning for collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 36(5):2069–2081, 2023.

A Appendix

We provide the comparison with NBFNet in Figure4.

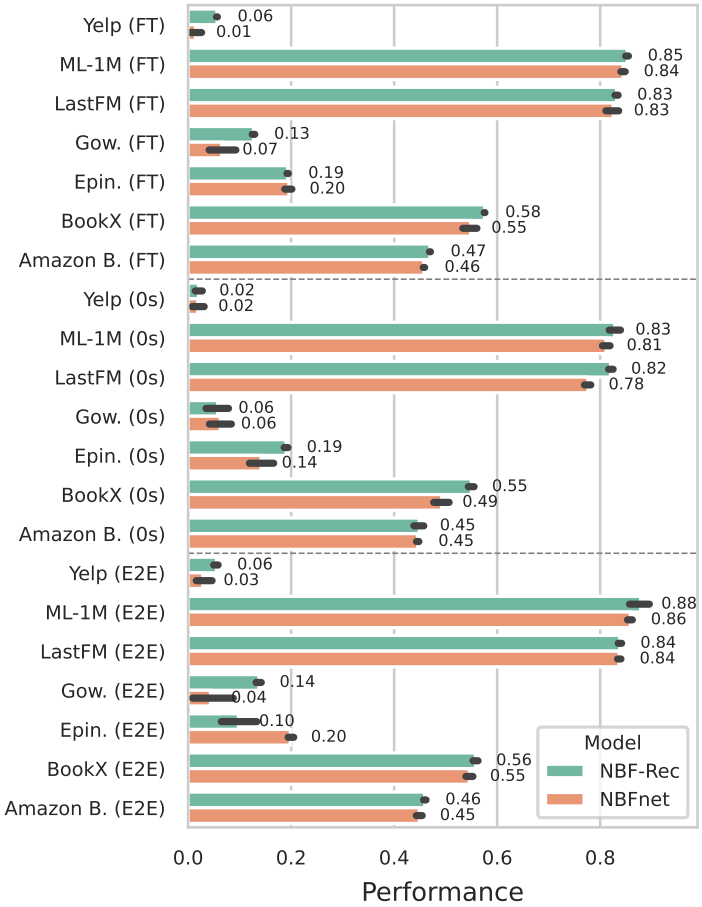

Figure 4: Performance comparison between NBF-Rec and NBFNet across all datasets under end-to-end, zero-shot, and fine-tuned settings. Incorporating interaction-level features and inductive training enables consistent improvements in recommendation quality.