Transferable Cross Domains Recommender System via Graph Fusion

Di Zhang, Qian Zhang, Jie Lu, Guanquan Zhang

Decision Systems and e-Service Intelligence Laboratory, AAII,
University of Technology Sydney, Sydney, NSW, Australia.

Abstract

Recommender systems has a very wide real world application in our everyday life. Graph neural networks (GNNs) have been widely adopted in recommendation systems due to their ability to capture user-item relationships in graph structures. However, data sparsity and cold start problems persist, especially when businesses expand into new sectors. GNN based recommender systems are often influenced by nodes bias, leading to over-fitting and over-smoothing issues as well as facing performance issues due to computational intensive algorithms. This paper proposes a novel framework that enhances target domain recommender system performance by adapting source domains user-item interactions via an augmented subgraph. (CDGF) We introduce a simple yet effective method to improve subgraph density, ultimately boosting performance while reducing graph structure complexity. Experiments conducted on Amazon review datasets demonstrate that our approach outperforms several well established methods across multiple baseline metrics under the data sparsity setting as well as pure cold start recommendation tasks.

Keywords: Data Sparsity; Cold-Start; Cross-Domains Recommendation; Graph Neural Networks; Graph Fusion

1

1 Introduction

Recommender systems have become essential in modern life, addressing the challenges posed by rapid data growth and constant product launches occurring every second. By helping users discover personalized products and services within an ever-expanding information landscape, these systems bring efficiency, focus, and simplicity to people's lives – making them high-demand technologies with wide-ranging applications and significant real-world impact.

For decades, Matrix Factorization (MF)-based [1] Collaborative Filtering (CF) [2] and Neural Factorization3 have dominated recommendation system approaches. These methods predict user preferences by analyzing user-item interactions, projecting both user and item features into a latent space where similarity is calculated through condensed vector representations. However, these approaches face well-documented limitations, particularly regarding cold start problems involving sparse data and new item recommendations. Common mitigation strategies like data warming techniques and frequent model retraining attempt to incorporate new interactions, but often prove inefficient, less timely, and costly to maintain.

Recent advances in recommender systems have seen growing adoption of graph-based approaches [3, 4]. Techniques like ode2Vec [5], DeepWalk [6], and Graph Convolutional Networks (GCN) [7] employ random walk strategies and neighborhood message passing to learn user/item representations that capture both local and global structural patterns. Notable applications include NGCF [8], LightGCN for general recommendations, social recommendation systems leveraging interpersonal connections [9], and knowledge graph-enhanced methods like KGAT. While these graph embedding methods demonstrate enhanced capability for handling continuously evolving information compared to traditional approaches, graph-based methods face several challenges including computational complexity, over-smoothing in deep architectures, and insufficient user/item data for effective representation learning.

Cross-domain recommendation has emerged as a promising solution for data sparsity challenges through knowledge transfer between domains. This approach leverages richer data from a source domain to enhance user/item representations in a data-scarce target domain, potentially improving recommendation performance. [10, 11] However, direct application of source domain data often proves suboptimal, as domain mismatch can introduce confounding patterns that reduce recommendation effectiveness, due to Divergent data distributions and structural patterns between domains, Inherited data sparsity issues in source domains, and noisy data that can negatively impact target domain performance.

We also noticed, most of the existing research had focused on explicit feedback, such as ratings, reviews, incorporate side information beyond the user-item interactions, such as user profiles, item features, etc. In hope of model can learn richer information beyond the implicit interactions. However, in many real-world scenarios, even under the age of LLM, extracting effective feature information remains a challenge, and belong to a dedicated research field. Not to mention, the prohibitive computational cost of processing text or multi-modal datasets, while preventing noisy pollution. Implicit feedback, such as click-through data, purchase history, are still the dominant or only training data points available for the recommender system model training.

In this paper we propose a novel framework to address the cold-start recommendation problem by leveraging the information from one or multiple source domains using only implicit feedbacks. We introduce a simple yet effective method boosting performance in the target domain subgraph argumentation, improving both subgraph density and reducing useless noisy information from source domain. Experiments conducted on Amazon review datasets demonstrate that our approach can tackle multiple real world recommendation challenges, including data sparsity, 0-interaction cold start scenarios.

To summaries, our contributions are as follows:

- We propose a novel recommendation framework (CDGF) that enhances target domain recommender system performance by adapting source domains via an augmented subgraph.
- We illustrate how CDGF can effectively improve target domain using only implicit interactions, implying the model can be applied to a wide range of real-world scenarios.
- Experiments are conducted against variations base line and metics shows promising effects beating well established recommendation models.

The rest of this paper is constructed as follows. In section 2, we discuss the related research and relevant definition that helped our research. In Section 3, we explain our framework and related algorithms. In Section 4, we show our experiment design and result compared with other baselines in cold-start scenario and sufficient data scenario. Lastly, we give conclusion and future directions in Section 5.

2 Related Works

This work is related to several research areas, including matrix factorization, graph neural networks, and cross-domain recommender systems. We provide an overview of these areas and discuss how our proposed method builds upon and extends existing approaches.

2.1 Matrices Factorization based Methods

Matrix Factorization (MF)-based and content-based recommender systems have long been foundational approaches in collaborative filtering. MF models, such as Bayesian Personalized Ranking (BPR) [12], optimize latent user-item interactions through pairwise ranking loss, enabling personalized recommendations by decomposing interaction matrices into low-dimensional embeddings. Extensions like Neural Collaborative Filtering (NCF) [13] integrate non-linear neural layers to capture complex user-item relationships. Content-based systems, in contrast, leverage item attributes (e.g., text, metadata) to recommend similar items, often using techniques like neural embeddings to model feature relevance. Hybrid approaches, such as wide and deep [14], unify collaborative and content-based signals by modeling feature interactions. Despite their effectiveness, traditional MF models struggle with cold-start scenarios and sparse data, while content-based methods may over-specialize recommendations.

2.2 Graph Neural Networks based Methods

Graph-based recommender systems leverage graph neural networks (GNNs) to model complex user-item relationships. GNNs aggregate node features and propagate information across graph structures, enabling embedding learning for users and items. Early models, like Graph Convolutional Networks (GCN) [7], use spectral convolutions to learn node representations. GAT [15] improves upon GCN by incorporating attention mechanisms to focus on relevant neighbors. GarphSAGE [16] and PinSage [17] use inductive learning to generalize to unseen nodes. Heterogeneous GNNs, like HetGNN [18], extend GNNs to multi-relational graphs, capturing diverse user-item interactions. Despite their flexibility, GNNs face challenges like scalability and over-smoothing, where node representations converge prematurely. LightGCN [19] simplifies GNNs by removing feature transformation and nonlinear activation functions, focusing on user-item interactions. Despite their effectiveness, GNNs struggle with the computational complexity of large-scale graphs and the over-smoothing issue. i.e. Nonlinear layers add parameters that may not align with the inherently user-item interactions, and Nonlinear transformations (e.g., ReLU) can distort collaborative signals[20], especially in shallow networks where linear aggregation suffices to capture user preferences. [21]

2.3 Cross Domains Recommender Systems

Cross-domain recommender systems (CDRs) address data sparsity and cold-start problems by transferring knowledge from a richer source domain to a sparser target domain. Traditional approaches, such as collective collaborative filtering is a direct extension of general MF based method. for example, CDCR [22] and DTCDR [23]. Developments leverage deep learning, including review-based methods that utilize textual side information to capture latent user-item interactions. such as, CoNet [24]. Graph based CDRs, like PPNG [10], leverage graph structures to propagate user preferences across domains. RecSys-DAN uses adversarial learning to learn user/item latent representations across domains.[11] However, challenges like negative transfer and non-overlapping sets persist. Fro CDRs, it also added the challenges on computational complexity and scalability, as the model needs to handle multiple domains with different distributions and structures. The over-smoothing issue also exists in CDRs, as the source domain data may not be optimal for the target domain, which may introduce unwanted noises.

3 Methodology

In this section, we describe the proposed method for cross-domain recommendation using graph neural networks. We first introduce the basic idea of the proposed method, followed by the graph construction for the target domain, source domain graph augmentation, and the adaptation of the source domain to the target domain. Finally, we discuss the model training and optimization process.
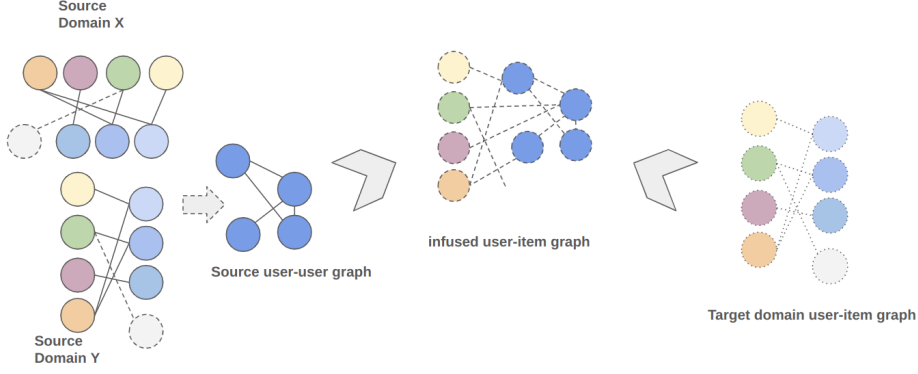
Fig. 1 Illustration of the proposed method: The source domain data is first augment from user-item graph into a user-user graph. Then the augmented source user-user graph is then adapted into the target domain as a subgraph. finally the model is trained on the infused graph in a coherent way.

3.1 Basic Idea

We propose a novel approach to address the cold-start recommendation problem by leveraging the information from one or multiple source domains using only implicit feedbacks.

Leveraging information from other domains to enrich user and item representations in the target domain. However, some source domains frequently suffer from data sparsity issues themselves, introducing unwanted noise and reducing the effectiveness of the source data. Hence in our approach, we focuses on both the effectiveness and efficiency when transferring information from the source domain to the target domain. We first structure the bipartite graph for target domain, where the user and item nodes are connected by the interactions. Next, we leverage graph augmentation on source domain data, the augmented graph brings a number of benefits:

- It can effectively improve graph density on shared nodes. (see more detailed comparison in section 4)
- Apply learnable edge regulation to adjust the edge weights based on the target domain data, which can effectively filter out the noises from the source domain data. e.g. reducing the impacts of highly connected nodes in the source domain, which may over weighted during the training process.
- user-user edges can be seamlessly integrated into the target domain graph, which can provide additional information for user representation learning.

We put adapt the augmented graph edges to the target domain by applying the edge regulation with a learnable weights. The edge regulation is designed to adjust the edge weights based on the target domain data, which can effectively filter out the noises from the source domain data. e.g. reducing the impacts of highly connected nodes in the source domain, which may over weighted during the training process.
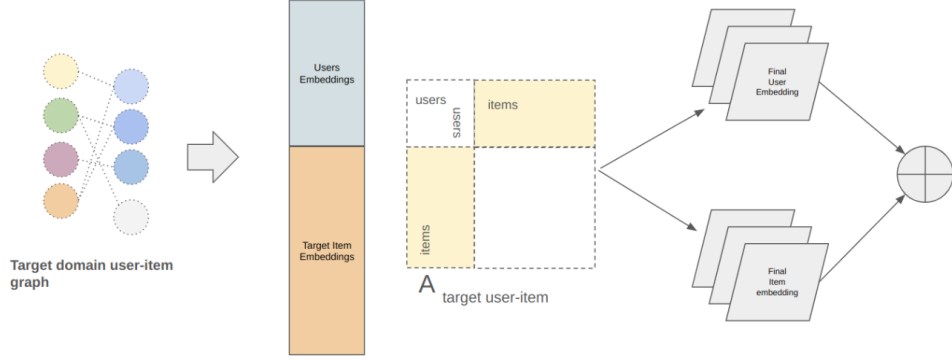
Fig. 2 Illustration of LightGCN aggregator on target domain.

Finally, source and target domain data would be put through a GNN model to learn the user and item representations. We use BPR as the optimization objective for training. The user and item representations are learned jointly, thus, the information from the source domain is used to improve the target domain recommendation performance. infuse target and source domains into a unified graph structure also improve the computational efficiency with less parameters to train, comparing to multi-graph approaches.

3.2 Graph Construction for Target Domain

We define the domain user-item interactions as:

$$G = (V_{user/item}, E_{interaction}) \tag{1}$$

Where $V_{user/item}$ are the user and item nodes, and $E_{interaction}$ are the interactions between users and items in target Domain.

User and Item nodes presentation are learned using Graph Neural Network (GNN) model via nodes messaging and layer aggregation.

The basic GNN model aggregation can be abstract as:

$$e_u^{(k+1)} = AGG(e_u^{(}k), e_i^{(}k), i \in N(u)) \tag{2}$$

AGG can be different implementations of Aggregator such as GCN [7, 15, 16] are commonly used in GNN models. The AGG function is the core of how message being propagated via nodes and its neighbors throughout the graph topology. Those aggregators are designed to learn the node representation by using non-linear activation functions in feature transformation. Its effectiveness is proven in many classifier tasks. However, when it comes to recommendation tasks, especially, when user interaction

6

are only training dataset available. The aggregator with nonlinearity functions commonly leads to over-smoothing issue, leads to poor performance and low variations in recommendation tasks [21].

On the other hand, the aggregator, such as, LightGCN [19], has proven results on improving the recommendation performance by adopting simple weighted sum aggregations, while reducing computation intensity at the same time. During our experiments, we also found that LightGCN being more superior handling the over-smoothing issue, comparing to former mentioned aggregators.

In our approach, we construct the target domain graph as following. we define a feature matrix for both user and item as E, while the adjacency matrix is defined as the user-item interaction matrix R=$R \in \mathbb{R}^{M \times N}$, where M and N are the user and item size. hence the adjacency matrix can be presented as:

$$E_{N+M} = \begin{pmatrix} E_{users} \\ E_{items} \end{pmatrix}, \quad A = \begin{pmatrix} 0, & R_{N \times M} \\ R_{M \times N}^T, & 0 \end{pmatrix} \tag{3}$$

hence, if we put the target domain through LightGCN aggregator:

$$E^{k+1} = (D^{-1/2} A D^{-1/2}) E^k \tag{4}$$

We can get final user and item representation e_{user}, e_{item} as:

$$e_{user} = \sum_{l=0}^{K} \alpha_l e_{user}^{(l)}; \quad e_{item} = \sum_{l=0}^{K} \alpha_l e_{item}^{(l)} \tag{5}$$

where K is the number of layers in the GNN model, and α_l is the learnable weights for each layer.

3.3 Source Domain Graph Augmentation

3.3.1 User-User Graph Augmentation

On the source domain side, we can derive the user-user or item-item graphs, based on common user/item interactions they share. (for simplicity, for the rest of the paper, we will use user-user graph as an example, the same logic can be applied to item-item graph as well.)

Say, the source domain has a user number of N' over lapping with the target domain $N \cap N' > 0$. The source user-user graph presented as User Feature Embedding E_s, and the user-user interaction matrix R_s.:

$$E_s = (E_{N'}), \quad A_s = (R_{N' \times N'}) \tag{6}$$

Apply degree normalization to the source user-user graph with weighted edges:

$$\tilde{A}_{s_i j} = \frac{w_{s_i j}}{\sqrt{d_{s_i}} \sqrt{d_{s_j}}} \tag{7}$$

The goal is remove the unuseable item nodes from the source. improving the density of the user-user graph, so the user similarity can be better learned and transferred to the target domain.
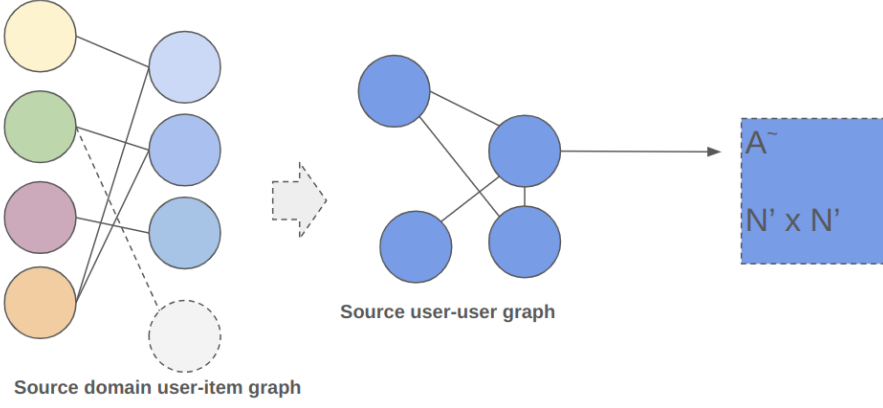
7

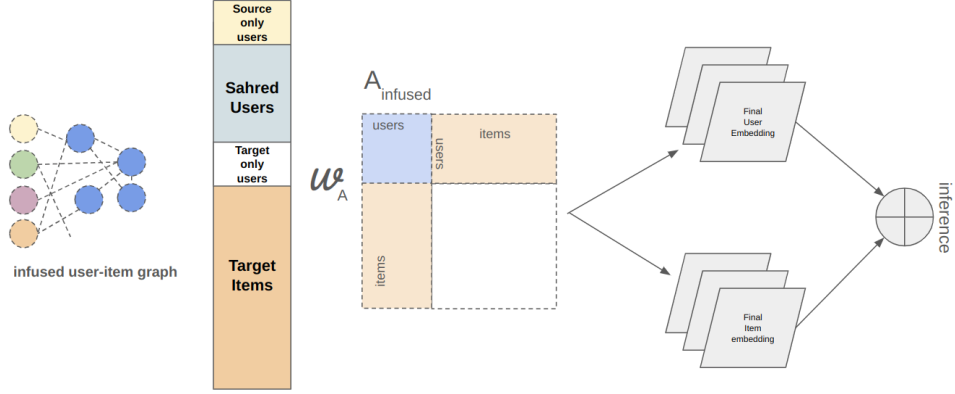**Fig. 3** Illustration of the source domain user-user graph.

Fig. 4 Illustration of the source domain graph infusion to the target domain.

3.3.2 Adapt Source Domain to Target Domain

We adapt the source user-user adjacency matrix to the target domain user-item adjacency matrix. Thus we get:

$$A_{infused} = \begin{pmatrix} 0 + \left(R_{N' \times N'}\right), & R_{N \times M} \\ R_{M \times N}^{T}, & 0 \end{pmatrix} \tag{8}$$

Source User Feature Embedding E_s will be shared with the target domain user feature embedding E_{user}, during training.

As we can see, during this process, fresh new users from the source, used to be out of target domain matrix, get introduced into the target domain from source. This enables the final model to make recommendation for those pure cold users, even

no user-item interaction data is available in the target domain. more details will be discussed in the experiment section.

3.4 Model Training and Optimization

3.4.1 Source Domain Transfer Comparison

Comparing to other source domain transfer methods, in general, we see the learning transfer is done via some form of dual or multi targets learning as model optimization objective.

A simplified loss function can be presented as:

$$
\begin{aligned}
\hat{y} &= f_{target}(x_{target}) \\
\hat{y}^{'} &= f_{source}(x_{source}) \\
L &= L_{target} + L_{source} + \lambda L_{reg}
\end{aligned}
\tag{9}
$$

where f_{target} and f_{source} are the target and source domain models, L_{target} and L_{source} are the loss functions for target and source domain, and L_{reg} is the regularization term.

CoNet [24] employs dual knowledge transfer across domains by introducing cross connections from one base network to another. Similarly, several other cross-domain recommender systems, such as PPNG [10] and DTCDR [23], adopt a similar approach with minor differences in their loss objectives, despite significant variations in their model architectures. In RecSys-DAN [11], transfer learning is achieved using a domain adversarial network.

A common aspect of these models above, are, all source embeddings are learned as part of the training process for knowledge transfer and domain alignment. Although these source domain embeddings are not used for the final recommendations, they still introduce additional parameters to train, increasing the model complexity.

3.4.2 Simplified Taining Objective

In our approach, we only use the source domain data to augment the target domain graph, hence no "waste" on the embedding learning for the model training. Thus, significantly reduces the model complexity and computational cost.

The trainable parameters in the model are the user and item embeddings, and the edge weights in the infused graph. the over lapped user between the source and target domain shares the same embeddings in the model.

We use Bayesian Probability Ranking as our optimization objective:

$$
L_{BPR} = - \sum_{(u,i,j) \in E} \log \sigma(e_u \cdot e_i - e_u \cdot e_j) + \lambda ||E||^2
\tag{10}
$$

Where σ is the sigmoid function, E is the set of user/item embeddings, e is the embedding out put from the final layer. the i and j are the positive and negative samples for user u. We use the L2 regularization, and the model is trained using Adam optimizer in mini batches.

9

4 Experiments and Results

First we will introduce the data set and experiment setup. Then we conduct experiments to evaluate the effectiveness through different scenarios and compare with other state-of-the-art base lines.

4.1 Dataset and experiment setup

We use amazon dataset for our experiments, which uses only user-item interactions as the input data. more specifically, we picked 2 product categories BOOK and MUSIC. between the 2 domains, we sampled 642 overlapping uniq users, see Table 1 for more details.

Domain	Users	Items	Interactions	Density
BOOK (Target Domain)	642	3022	8857	0.00456
MUSIC (Source Domain)	642	3507	12434	0.00641

Table 1 Dataset statistics

4.1.1 Deriving a denser user-user graph from source domain

Domain	Node number	Edge number	Density
Source user-item	4149	8857	0.00456
Source user-user	642	46952	**0.11391**

Table 2 Source domain user-user graph statistics

We argument the source domain user-item graph to user-user graph. The process, not only simplifies the source graph. the end results also significantly improves the density of the user-user graph. while, it also effectively reduced the noises by discard source items nodes. The fact that, the item feature is no use for the target domain recommendation. This also reduces the computational cost of the model. see Table 2 for more details.

4.2 Baseline comparison

Fro the baseline comparison, we compare the following algorithms:

- **BPR**: Bayesian Personalized Ranking
- **LightGCN**: Simplifying and Powering Graph Convolution Network for Recommendation
- **CoNet**: Deep Dual Transfer Cross Domain Recommendation

We use the following evaluation metrics to evaluate the performance of the recommendation algorithms:

- **Precision@k** (P@k): the proportion of recommended items in the top-k that are relevant to the user.
- **Recall@k** (R@k): the proportion of relevant items in the top-k that are recommended to the user.
- **NDCG@k**: Normalized Discounted Cumulative Gain, which is a measure of ranking quality.

4.3 Results

We compare the performance of the proposed method with the baseline algorithms in terms of Precision@25, Recall@25, and NDCG@25. The results are shown in Table 3.

Algorithm	Precision@25	Recall@25	NDCG@25
BPR	0.002852	0.017115	0.009939
LightGCN	0.005445	0.036747	0.024399
CoNet	0.003122	0.002123	0.016799
OURS	**0.007714**	**0.048557**	**0.028301**

Table 3 Results comparison

5 Conclusion and Future Work

In this paper, we propose an infused graph framework for recommendation (CDGF). This framework offers a scalable and efficient solution suitable for large-scale recommendation systems. It supports various domain overlapping scenarios, including both partial and full overlapping data conditions on either common users or items. Additionally, it can handle multiple domains with different distributions and structures and can be easily extended to incorporate more domains.

We evaluated the effectiveness of the model on real-world datasets and demonstrated that it outperforms state-of-the-art methods in terms of recommendation accuracy, particularly effective in handling sparse data and cold-start scenarios, while using only implicit user-item interactions. Importantly, in an era where data privacy is a growing concern, our model achieves high-quality recommendations without relying on personally identifiable information (PII) or requiring complex data engineering efforts.

Future work will focus on extending the model to handle more complex scenarios, such as enhancing its inference ability to make transductive predictions and managing dynamic data changes within the graph architecture. These improvements will further enhance the model's accessibility and ultimately aid in the adoption of real-world recommender systems.

References

[1] Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009)

[2] Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) **22**(1), 5–53 (2004)

[3] Mao, M., Lu, J., Zhang, G., Zhang, J.: Multirelational social recommendations via multigraph ranking. IEEE Transactions on Cybernetics **47**(12), 4049–4061 (2016)

[4] Wang, W., Zhang, G., Lu, J.: Member contribution-based group recommender system. Decision Support Systems **87**, 80–93 (2016)

[5] Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864. ACM, ??? (2016)

[6] Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining **20**, 701–710 (2014)

[7] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 **1** (2016)

[8] Wang, X., He, X., Wang, M., Feng, F., Chua, T.-S.: Neural graph collaborative filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 165–174 (2019)

[9] Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: meta path-based top-k similarity search in heterogeneous information networks. Proceedings of the VLDB Endowment **4**(11), 992–1003 (2011)

[10] Zhao, C., Li, C., Fu, C.: Cross-domain recommendation via preference propagation graphnet. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 2165–2168 (2019)

[11] Wang, C., Niepert, M., Li, H.: Recsys-dan: Discriminative adversarial networks for cross-domain recommender systems. IEEE transactions on neural networks and learning systems **31**(8), 2731–2740 (2019)

[12] Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618 **12** (2012)

[13] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.-S.: Neural collaborative

filtering. In: Proceedings of the 26th International Conference on World Wide Web, pp. 173–182 (2017). International World Wide Web Conferences Steering Committee

[14] Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., *et al.*: Wide & deep learning for recommender systems. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp. 7–10. ACM, ??? (2016)

[15] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph Attention Networks (2018). https://arxiv.org/abs/1710.10903

[16] Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems, vol. 30, pp. 1024–1034 (2017)

[17] Pal, A., Eksombatchai, C., Zhou, Y., Zhao, B., Rosenberg, C., Leskovec, J.: Pinnersage: Multi-modal user embedding framework for recommendations at pinterest. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2311–2320 (2020)

[18] Shi, C., Hu, B., Zhao, W.X., Philip, S.Y.: Heterogeneous information network embedding for recommendation. IEEE Transactions on Knowledge and Data Engineering **31**(2), 357–370 (2018)

[19] He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: 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, pp. 639–648. ACM, ??? (2020)

[20] Wei, X., Liu, J.: Effects of nonlinear functions on knowledge graph convolutional networks for recommender systems with yelp knowledge graph. (2021). https://api.semanticscholar.org/CorpusID:236406259

[21] Sharma, S., Paranjape, V.: Experimental hybrid approach for improving the quality of personalized product recommendation systems with deep learning. International Journal of Innovative Research in Computer and Communication Engineering (2023)

[22] Rafailidis, D., Crestani, F.: A collaborative ranking model for cross-domain recommendations. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 2263–2266 (2017)

[23] Zhu, F., Chen, C., Wang, Y., Liu, G., Zheng, X.: Dtcdr: A framework for dual-target cross-domain recommendation. In: Proceedings of the

28th ACM International Conference on Information and Knowledge Management. CIKM '19, pp. 1533–1542. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3357384.3357992 . https://doi.org/10.1145/3357384.3357992

[24] Hu, G., Zhang, Y., Yang, Q.: Conet: Collaborative cross networks for cross-domain recommendation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. CIKM '18, pp. 667–676. ACM, ??? (2018). https://doi.org/10.1145/3269206.3271684 . http://dx.doi.org/10.1145/3269206.3271684