Time Distance Aware for Multi-component Graph Collaborative Filtering

Tseesuren Batsuuri✉[0009−0000−4489−0787], Shan Xue, Jian Yang, and Jia Wu

School of Computing, Macquarie University, Sydney, NSW, Australia
`tseesuren.batsuuri@hdr.mq.edu.au`, `{emma.xue,jian.yang,jia.wu}@mq.edu.au`

Abstract. Graph Convolutional Networks (GCNs) have gained prominence in collaborative filtering (CF) recommendation systems for capturing intricate signals using high-order structural data. However, GCN-based models focus solely on these signals, neglecting the sparse nature of data and overlooking important aspects like temporal signals in user preferences and baseline signals in users or items, leading to sub-optimal performance. To address these issues, this paper introduces a novel multi-component CF model that integrates GCNs with baseline and temporal components. The integrated model learns user and item representations from multiple perspectives, enhancing performance and robustness across various datasets. Experiments conducted on the MovieLens and Douban datasets demonstrate the superiority of this approach over state-of-the-art models, reducing RMSE by up to 4.7%, while improving NCDG by up to 5.1% compared to pure GCN-based CF (see source code[1]).

Keywords: collaborative filtering · multi-component learning · recommender system · graph convolutional networks · temporal data.

1 Introduction

CF is a fundamental technique in recommendation systems, standing alongside content-based filtering. One key advantage of CF is its minimal reliance on extensive data preparation, effectively leveraging user-item interaction data. It operates without domain knowledge, instead harnessing similarities between users and items derived from their past interactions to formulate recommendations. Model-based CF uses statistical models [1, 2] and has evolved through three generations: shallow, neural, and graph neural models [2]. Shallow models like SVD, SVD++ [3] relied on matrix factorization, where latent features representing users and items are learned and used in a simple "dot product" to predict ratings. Neural models, such as Nade-CF, AutoRec, and SparseFC [4–6], have enhanced CF by leveraging deeper neural networks. Recently, GCNs have emerged as a significant advancement in this field [7–11], providing the ability to capture intricate signals using high-order structural data. GCNs are a natural fit for CF [7, 8, 10] because CF predicts a user's preferences based on the

---

[1] `https://github.com/tseesurenb/wise2024_v2.git`

collective behavior and preferences of a group of users (or items). GCNs aggregate signals from nodes connected by shared features, aligning well with the fundamental principles of CF. However, GCN-based models still face challenges such as data sparsity [1, 2] and incorporating the evolving nature of user preferences [3, 12, 15]. Particularly, temporal dynamics in user preferences are often overlooked [1,3,12], leading to sub-optimal models. User preferences are intricate and evolve over time, making it crucial for recommendation systems to adapt to these changes. Temporal effects in user preferences can be viewed at three levels: immediate, transient, and long-term [3, 12]. Modeling immediate temporal effects is essentially impossible due to their highly subjective nature [3]. The long-term temporal effect has been studied in the classical SVD++ [3], where inherent baseline signals, or biases, represent the long-term view, capturing general tendencies of users or items. For example, biases can indicate whether a user's ratings are generally optimistic or pessimistic.

The most intriguing aspect of this research is modeling transient temporal effects, which can be classified into two key types: relative and absolute.

Relative: Like Avatar left a lasting mark on films, the first iPhone changed how we see phones. But in everyday life, recent experiences matter more—when buying a new smartphone, people usually compare it to their last one.

Absolute: Following prestigious events such as the Oscars or Golden Globes, individuals often rate films higher, as high-quality movies are typically released during these periods. This illustrates the concept of absolute temporal effects.

To address these challenges, this paper introduces a novel multi-component CF approach called TDMCF, which integrates GCNs with baseline and temporal components using supervised learning for explicit data. By incorporating these elements, our model captures transient shifts in user preferences and remains robust against sparse data. Our contributions are threefold:

- **Novel integrated CF**: We present a novel CF model that integrates GCNs with baseline and temporal components. This model captures high-order collaborative signals, incorporates baseline signals to learn user and item biases, and models user dynamics through temporal components.
- **Two Novel Time Modules for CF**: We propose two novel time modules to capture dynamic changes in user preferences: the relative time module adjusts past interactions based on user-specific timestamps, while the absolute time module standardizes temporal influence using a global minimum timestamp. These modules enhance collaborative filtering by better reflecting evolving user preferences.
- **Experimental Validation**: Experiments conducted on the MovieLens and Douban datasets demonstrate the superiority of our approach over state-of-the-art models.

The remainder of the paper is organized as follows: Section 2 presents the problem statement with preliminary concepts. Section 3 introduces our proposed model, and Section 4 presents the experiments and results. Section 5 provides a literature review, and finally, Section 6 concludes the paper.

2 Preliminary and Problem statement

Let $U = \{u_1, ..., u_N\}$ be a set of N users, and $I = \{i_1, ..., i_M\}$ be a set of M items. In our scenario, interactions between the users and items are given in a log, represented by list of four-tuples, denoted as $\{(u, i, r_{ui}, t_{ui})\}$, where $r_{ui} \in \{1, ..., K\}$ with a K scale is a rating given by user u to item i at time t_{ui}.

To capture collaborative signals, we require embeddings e_u, e_i for every user and item and an adjacency matrix A of a bipartite graph $G = (V, E)$. V is a union set $U \cup I$, and E is the set of edge triples (u, i, r_{ui}). The embeddings with dimension of d and adjacency matrix $A \in \mathbb{R}^{(N+M)\times(N+M)}$ are defined as:

$$H_c = [\underbrace{e_{u_1}, ..., e_{u_N}}_{\text{user}}, \underbrace{e_{i_1}, ..., e_{i_M}}_{\text{item}}], \quad A = \begin{bmatrix} 0 & R \\ R^T & 0 \end{bmatrix} \tag{1}$$

where $e \in \mathbb{R}^d$, $R \in \mathbb{R}^{N\times M}$ is the interaction matrix, and 0 is all-zero matrix.

To capture the baseline signals, embeddings b_u, $b_i \in \mathbb{R}^d$ for every user and item, and a global mean rating μ are defined as:

$$H_b = [\underbrace{b_{u_1}, ..., b_{u_N}}_{\text{user baseline}}, \underbrace{b_{i_1}, ..., b_{i_M}}_{\text{item baseline}}], \quad \mu = \frac{\sum r_{ui}}{len(r_{ui})} \tag{2}$$

To capture the transient temporal signals, two more embeddings are defined for every user: relative and absolute temporal embeddings t_u^r, $t_u^a \in \mathbb{R}^d$:

$$H_t = [\underbrace{t_{u_1}^r, ..., t_{u_N}^r}_{\text{relative}}, \underbrace{t_{u_1}^a, ..., t_{u_N}^a}_{\text{absolute}}] \tag{3}$$

The problem statement: With separate embeddings for users and items in Eqs. (1)-(3), TDMCF is designed to leverage collaborative, baseline and temporal signals captured from the interaction log so as to predict the missing ratings. The adjacency matrix A together with embeddings H_c serves as input for the GCN component of the model. The global mean rating μ together with embeddings H_b serves as input for the baseline component of the model for capturing user and item biases. The relative and absolute temporal sets and their embeddings H_t serve as input for a temporal component of the model.

3 TDMCF: Time Distance aware Multi-Component CF

We defined two tenets to model evolving user preference. First, user preference is not inherent, rather, they develop through past experiences with user judgments being primarily dependent on comparing the current experience to previous ones. Second, different experiences exert varying degrees of lasting influence. Given these tenets, we defined two time models. One is a relative time model:

$$p_u^r = f(t_u^r, t_u - t_{u_{\min}}) = t_u^r \cdot e^{(-\beta^r \cdot (t_u - t_{u_{\min}}))} \tag{4}$$

where β^r is a hyper-parameter, $t_{u_{\min}}$ is the relative starting point, and in our case, it is the minimum time of the user. The other one is an absolute time model:

$$p_u^a = f(t_u^a, t_u - t_{\min}) = t_u^a \cdot (t_u - t_{\min})^{\beta^a} \tag{5}$$

where β^a is a hyper-parameter, $t_{\min}$ is the fixed initial point, and in our case, it is the global minimum time in the interaction log.

3.1 Simplified GCN for deeper collaborative signals

As stated, TDMCF model has three components and the core component is the GCN component and we chose LightGCN [10] for two reasons. First, it is specifically proven simplified CGN for CF. Second, LightGCN is a foundational, feature-less that is ideal for other models to build upon as shown in Eq. 6.

$$\underbrace{e_u^{k+1} = \sum_{i \in N_u} \frac{1}{\sqrt{|N_i|}\sqrt{|N_u|}} e_i^k}_{\text{(1) User embedding update}}, \quad \underbrace{e_i^{k+1} = \sum_{u \in N_i} \frac{1}{\sqrt{|N_i|}\sqrt{|N_u|}} e_u^k}_{\text{(2) Item embedding update}} \tag{6}$$

where N_u, N_i are the node degrees, e_u, e_i are embeddings of user u and item i, and k and $k+1$ are the k^{th} and $(k+1)^{th}$ layers, respectively. To obtain the final embeddings, LightGCN sums over the embeddings, as described in Eq. 7.

$$\underbrace{e_u = \sum_{k=0}^{K} \alpha_k e_u^{(k)}}_{\text{(1) Aggregation of user embeddings}} \quad , \quad \underbrace{e_i = \sum_{k=0}^{K} \alpha_k e_i^{(k)}}_{\text{(2) Aggregation of item embeddings}} \tag{7}$$

where $\alpha_k \geq 0$ denotes the weight parameter.

3.2 Multi-Component CF

As shown in Figure 1, the proposed multi-component TDMCF model has three components: the baseline, the temporal and the collaborative. All three components contribute to the final prediction defined as:

$$r_{ui} = g(\underbrace{\mu + b_u + b_i}_{\text{baseline}} + \underbrace{p_u^r + p_u^a}_{\text{temporal}} + \underbrace{e_u^T \cdot e_i}_{\text{collaborative}}) \tag{8}$$

where r_{ui} is the final predicted rating, $g(\cdot)$ is an activation function such as ReLU, μ is the global mean rating, b_u, b_i are user and item baseline signals, p_u^r is the relative temporal signal given by Eq. (4), p_u^a is the absolute temporal signal given by Eq. (5) and $e_u^T \cdot e_i$ is the collaborative signal given by Eq. (7).

The TDMCF is optimized using the "Adam" optimizer to minimize the mean squared error (MSE) with the following loss function:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{M} \sum_{i=1}^{M} \|r_m - \hat{r}_m\|_2^2 + \lambda \|w\|_2^2 \tag{9}$$

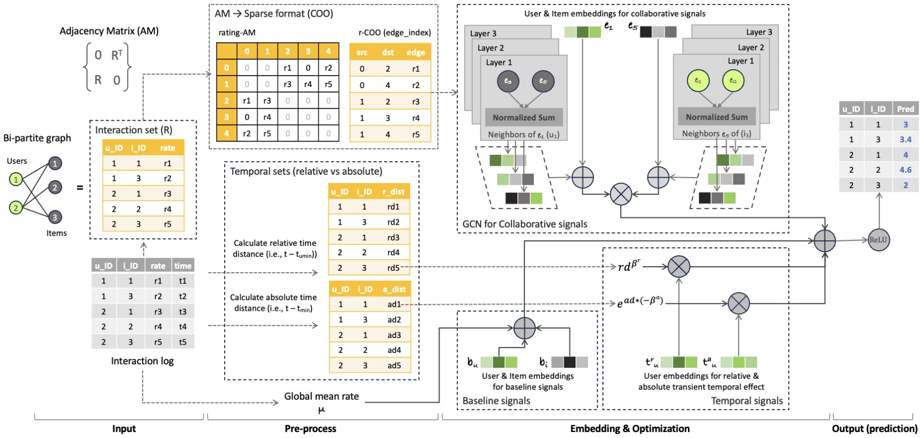

Fig. 1. The interaction log is preprocessed into an interaction set and temporal sets. These sets are fed into TDMCF, which predicts missing ratings using its baseline, temporal, and collaborative components.

Here, M denotes the number of samples in a batch, and r_m and $\hat{r}_m$ signify the predicted and target values for the m-th sample. The parameter λ is the weight decay factor, and $|w|_2^2$ signifies the L2 norm of the model's weight vector w.

4 Experiments and Results

We used the MovieLens ml-100k and ml-1m datasets [2], along with the Douban-book dataset [18]. Each user in these datasets has rated at least 20 items (for the Douban-book, we preprocessed the data to ensure each user has at least 20 ratings), with each rating including a timestamp. Table 1 presents the statistics for each dataset. As shown, the Douban-book dataset is extremely sparse. The experiments were conducted using 90% training and 10% test sets, and average results with standard deviations were computed. We split the dataset without regard to time order, as the proposed model aims to learn parameters based on

Table 1. Statistics of the experimental datasets

Dataset	Items	Users	Ratings	Sparsity	Graph Density	Duration
MovieLens 100k	1682	943	100,000	93.7%	2.9%	215 days
MovieLens 1M	3952	6040	1,000,209	95.83%	2.0%	1039 days
Douban-book	1403	94651	221,195	99.8%	0.005%	4797 days

---

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

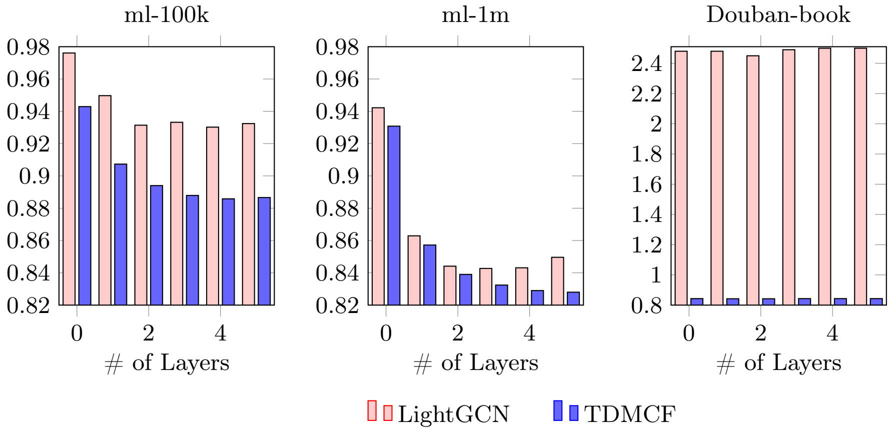

Fig. 2. The TDMCF significantly enhances RMSE and effectively addresses the over-smoothing with GCNs. Note, the pure GCN model performs poorly on the Douban-book dataset due to its highly sparse data, with an average node degree of 4.6. In contrast, the other datasets have average node degrees of 76.2 and 200.2, respectively.

time distance. The only concern is to ensure that the minimum timestamp of the dataset and the minimum timestamp for each user are included in the training set. The models were trained with a learning rate of 1e-3, weight decay ranging from 1e-5 to 1e-6, and vector dimensions set to 64 for ml-100k, Douban-book and 500 for ml-1m datasets. The parameters β^r and β^a were both set to 0.05 and 25 for movielens datasets and 0.001 to 0.008 for douban-book dataset, respectively. For TDMCF, it is crucial to train the model with a larger batch size or even the entire training set. This is due to the time-distance training component.

4.1 Results

As depicted in Figure 2 and 3, TDMCF clearly demonstrates superior performance. It achieves a notable 4.7% reduction from (0.93 to 0.886) in RMSE on the ml-100k and 1.7% reduction (from 0.8427 to 0.8284) on the ml-1m compared to LightGCN. Additionally, TDMCF enhances the NCDG by 1.8% (from 0.7958 to 0.81) on the ml-100k and 0.4% (from 0.8228 to 0.826) on the ml-1m. On the Douban-book dataset, the model produces even more intriguing results. For this dataset, a pure GCN model simply fails, with an RMSE of around 2.4, indicating a significant error as shown in Figure 2 (adding more layers did not help). The Douban-book dataset has limited high-order structural data, with an average node degree of only 4.6, compared to 76.2 for ml-100k and 200.2 for ml-1m. On the contrary, the TDMCF performs well on this dataset as highlighted in Figure 2 and 3.

Finally, Table 2 presents the results of TDMCF vs. baseline methods. We selected two classic CFs and three state-of-the-art neural models, as well as six state-of-the-art GCN-based approaches. The TDMCF outperforms all state-of-

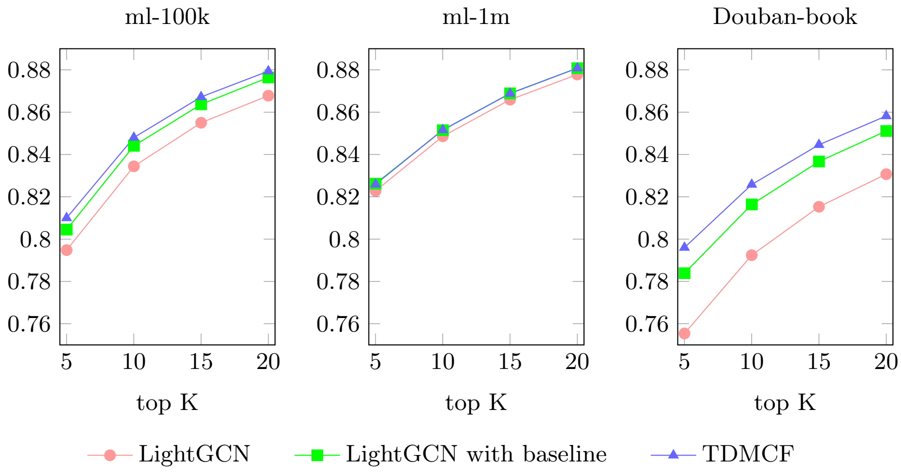

Fig. 3. The TDMCF enhances NCDG for all three datasets. Especially, for extreme sparse dataset, namely, Douban-book, it is clearly demonstrated.

the-art baseline methods on the ml-100k dataset. On the ml-1m dataset, TDMCF remains the strongest contender, with the exception of SparseFC.

Table 2. The TDMCF yields state-of-the-art results.

Methods	**RMSE (ml-100k)**	**RMSE (ml-1m)**
SVD++ [17]	0.903	0.856
timeSVD++ [3]	0.894	0.847
GraphRec [16]	0.898	0.867
GHRS [17]	0.887	0.838
TG-MC with GRU [15]	-	0.867
TG-MC with LSTM [15]	-	0.834
I-AutoRec [4]	0.895	0.831
IMC-GAE [14]	0.897	0.829
CF-Nade [5]	-	0.829
SparseFC [6]	0.89	0.824
GC-MC [7]	0.905	0.832
LightGCN	0.930	0.843
TDMCF	**0.886**	**0.828**

Discussion: First, over-smoothing is a significant concern. After a certain number of layers, GCNs tend to make almost all node embeddings similar, leading to a phenomenon known as over-smoothing. This reduces the model's ability to distinguish between different nodes and can negatively impact performance.

Second, the average node degree is crucial for the success of GCN-based CF. While GCNs leverage high-order structural data, allowing nodes to aggregate information from their neighbors, this becomes problematic in extremely sparse datasets, such as the Douban-book dataset, where the average node degree is just 4.6. In such cases, these higher-order connections are less effective, directly contributing to poor performance. Given these conditions, it is essential to adopt an integrated solution like TDMCF, which incorporates multiple components to simultaneously capture patterns from various perspectives.

Third, our experiments reveal the impact of noisy interactions. While performance generally improves with an increasing number of layers, we also observe instances of performance degradation followed by recovery. This inconsistency can be attributed to noisy neighbors. Current GCN models for collaborative filtering treat all nodes equally, but some nodes introduce noise. A noisy item with a high node degree can disproportionately affect recommendations, as message aggregation spreads this noise. This highlights an opportunity for future work.

Ablation Study: Table 3 presents the breakdown of TDMCF's components. These include the baseline-only (i.e., *base*), baseline + absolute time model (i.e., $base + abs$), baseline + relative time model (i.e., $base + rel$), and the full model. Both time models, in addition to the baseline, contribute proportionally to the performance imporvements. We also conducted experiments with temporal embeddings without multiplying them by time distance functions (see abs^{raw} and rel^{raw}). Compared to the baseline-only, these embeddings alone not only failed to improve the performance, they actually made it worse.

5 Related Work

As GCNs have emerged as the latest advancement in CF, we first studied GCN-based CF models and then examined temporal CF models.

Graph Convolutional Matrix Completion (GC-MC) [7] is a pioneering approach that uses a GCN for CF, integrating a GCN into an auto-encoder architecture for matrix completion tasks. However, GC-MC primarily focuses on

Table 3. The RMSE performance breakdown of TDMCF by components.

Components	**ml-100k**	**Drop (%)**	**ml-1m**	**Drop (%)**
LightGCN	0.9302	-	0.8427	-
TDMCF				
- $base$	0.9025	-2.98%	0.8300	-1.51%
- $base + abs$	0.9009	-3.15%	0.8290	-1.63%
- $base + rel$	0.8969	-3.58%	0.8294	-1.58%
- $full$	0.8858	-4.78%	0.8284	-1.7%
- $base + abs^{raw}$	0.904	+0.10%	0.8300	+0.00%
- $base + rel^{raw}$	0.902	-0.02%	0.8300	+0.00%

static collaborative signals, lacking the ability to account for temporal dynamics in user preferences, leading to suboptimal performance over time.

Neural Graph Collaborative Filtering (NGCF) [8] enhances embeddings by incorporating high-order connectivity relations through multiple embedding propagation layers. While effective in capturing structural data, NGCF overlooks temporal aspects and user-item interaction biases, limiting its adaptability to changing user preferences. LightGCN [10] simplifies GCNs by focusing on message aggregation and removing feature transformation and non-linear activation. This approach addresses computational complexity but does not account for temporal dynamics or baseline biases, which are crucial for capturing evolving user preferences. A Linear Residual - Graph Convolutional Collaborative Filtering (LR-GCCF) model [11] also addresses the over-smoothing issue in GCNs by introducing residual connections. Although it improves performance by mitigating over-smoothing, LR-GCCF does not integrate temporal dynamics, reducing its effectiveness in modeling dynamic user preferences.

Despite GCNs being the latest advancement in CF, we argue that GCN-based models primarily focus on capturing deeper collaborative signals. However, collaborative signals alone are insufficient to fully model evolving user preferences. Additionally, GCNs are effective only with datasets that have high-order structural data and may perform poorly when such information is lacking.

5.1 Temporal collaborative filtering

Incorporating temporal data into CF has garnered relatively less attention than other aspects [1]. Nonetheless, the first noteworthy attempt is evident in [3]. This work proposes a comprehensive approach for capturing both long-term and transient patterns. Integrating a temporal model with classic memory-based and SVD++ models significantly improves the accuracy. In addition to [3], the more recent study [15] introduces an intriguing temporal CF that combines a GCN and recurrent neural networks (RNN). They use GC-MC [7] as a base GCN model. To model the temporal part, they have opted for a gated recurrent unit (GRU) and a long short-term memory (LSTM). The concept is clear, seeking to combine GCN's strength with high-order structural data and RNN's proficiency in handling temporal data. The only evident drawback is the framework's increased complexity and the necessity for a larger neural networks.

6 Conclusion

This paper introduces TDMCF, a novel integrated CF model with three components: a GCN component for capturing deeper collaborative signals, a temporal component for tracking evolving user preferences, and a baseline model for addressing user and item biases. This integrated approach enhances traditional GCN-based CF, improving accuracy and robustness across various datasets.

References

1. Wang, S., Hu, L., Wang, Y., He, X., Sheng, Q.Z., Orgun, M.A., Cao, L., Ricci, F., Yu, P.S.: Graph Learning based Recommender Systems: A Review. In: International Joint Conference on Artificial Intelligence, pp. 4644–4652. IJCAI (2021)
2. Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., Quan, Y., Chang, J., Jin, D., He, X., Li, Y.: A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions. ACM Transactions on Recommender Systems. ACM, New York, NY, USA (2023)
3. Koren, Y.: Collaborative Filtering with Temporal Dynamics. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 447–456. ACM, New York, NY, USA (2009)
4. Sedhain, S., Menon, A.K., Sanner, S., Xie, L.: AutoRec: Autoencoders Meet Collaborative Filtering. In: Proceedings of the 24th International Conference on World Wide Web, pp. 111–112. ACM, New York, NY, USA (2015)
5. Zheng, Y., Tang, B., Ding, W., Zhou, H.: A neural autoregressive approach to collaborative filtering. In: International Conference on Machine Learning, pp. 764–773. PMLR (2016)
6. Muller, L., Martel, J., Indiveri, G.: Kernelized synaptic weight matrices. In: International Conference on Machine Learning, pp. 3654–3663. PMLR (2018)
7. Van den Berg, R., Kipf, T.N., Welling, M.: Graph Convolutional Matrix Completion. In: Proceedings of the 24th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 433–442. ACM, London, UK (2017)
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. ACM (2019)
9. Sun, J., Zhang, Y., Ma, C., Coates, M., Guo, H., Tang, R., He, X.: Multi-graph convolution collaborative filtering. In: 2019 IEEE International Conference on Data Mining, pp. 1306–1311. IEEE (2019)
10. 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, New York, NY, United States (2020)
11. Chen, L., Wu, L., Hong, R., Zhang, K., Wang, M.: Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In: Proceedings of the AAAI conference on artificial intelligence, vol. 34, no. 01, pp. 27–34. AAAI (2020)
12. Liu, K., Xue, F., Guo, D., Wu, L., Li, S., Hong, R.: MEGCF: Multimodal Entity Graph Collaborative Filtering for Personalized Recommendation. In: Proceedings of the ACM conference, vol. 41, no. 2, article no. 30, pp. 1–27. ACM (2023).
13. Wu, F., Souza, A., Zhang, T., Fifty, C., Yu, T., Weinberger, K.: Simplifying graph convolutional networks. In: International Conference on Machine Learning, pp. 6861–6871. PMLR (2019)
14. Shen, W., Zhang, C., Tian, Y., Zeng, L., He, X., Dou, W., Xu, X.: Inductive Matrix Completion Using Graph Autoencoder. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, pp. 1609–1618. ACM, NY, USA (2021)
15. Bonet, E.R., Nguyen, D.M., Deligiannis, N.: Temporal Collaborative Filtering with Graph Convolutional Neural Networks. In: Proceedings of 25th International Conference on Pattern Recognition, pp. 4736–4742. IEEE (2021)

16. Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J., Yin, D.: Graph Neural Networks for Social Recommendation. In: Proceedings of WWW 2019: International World Wide Web Conferences, pp. 417–426. ACM, New York, NY, USA (2019)
17. Zamanzadeh Darban, Z., Valipour, M.H.: GHRS: Graph-Based Hybrid Recommendation System with Application to Movie Recommendation. Expert Syst. Appl. (2022)
18. Zhu, F., Wang, Y., Chen, C., Liu, G., Zheng, X.: A graphical and attentional framework for dual-target cross-domain recommendation. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, article no. 415, pp. 1–8. IJCAI, Yokohama, Japan (2021)