

UniDNR: A Novel Interaction Graph and Texts Fusion Method for Review-based Recommendation

Yu-Qiao Liu^{1,2}, Nan Zheng^{1,2(✉)}, and Song Zhang^{1,2}

¹ School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

² State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, 100190, China
{liyuyqiao2022, nan.zheng, zhangsong2022}@ia.ac.cn

Abstract. Review-based recommender systems aim to calculate users' preference for items by leveraging user reviews. Current methods mainly consist of two components: user and item embedding learning and user-item rating predicting. But these methods overlook the higher-order interaction relationships in the user-item graph which are beneficial to capture users' preferences and features among items. Also, these methods overlook the inherent attributes in item descriptions which complement user reviews. In this paper, we propose a deep neural recommendation framework named UniDNR that unites item descriptions, user reviews and the user-item interaction graph to make recommendations. UniDNR can be divided into three parts: the ID-level embedding layer, the text-level embedding layer and the rating prediction layer. Specifically, the ID-level embedding layer captures the higher-order interactive relationship in the user-item interaction graph which can better share features among users and items. The text-level embedding layer focuses on embedding items and users by aspect-based learning which considering different aspects mentioned in descriptions and reviews. Such that, we combine ID embedding and text embedding to predict the most likely final rating assigned by the user. Experiments on three real-world datasets demonstrate the superiority of our proposed UniDNR model compared to the state-of-the-art baselines.

Keywords: Recommender systems · Text information · Neural network

1 Introduction

Recommender systems have been widely applied in various fields, such as e-commerce[1], social media[2], and so on. With the rapid development of natural language processing technology, more and more methods are available for analyzing textual data such as user reviews[3-5]. So many approaches leverage user reviews to enhance the performance of recommender systems[6-9]. Most of the existing review-based recommender systems[6, 9, 10] employ two modules to predict user preferences for items: (1) user and item embedding representing, and (2) user-item rating predicting. For example, two-tower neural networks[9] are widely used in review-based

recommender systems, where two encoders are employed to learn representations of users and items from reviews, respectively. Then it employs factorization machine(FM) [11] to predict user ratings for items.

Despite many advantages of review-based recommendation methods, there are still some drawbacks. Firstly, as shown in Figure1 (a), the historical interaction data of users and items can naturally form a user-item interaction graph, but current methods overlook its higher-order interactive relationship. On one hand, recommender systems can capture users' preferences and behaviors by analyzing the higher-order interaction relationship between users and items. On the other hand, the higher-order interactive relationship allows recommender systems to better share features among items which interact indirectly. Secondly, user reviews are typically subjective, influenced by individual preferences and emotions. Item descriptions, often crafted by professionals, tend to be more objective and consistent. As shown in Figure1 (b), item descriptions typically include the inherent attributes of items which complement user reviews. To this end, in this paper, we propose a deep neural recommendation model named UniDNR that unites item descriptions, user reviews and the user-item interaction graph to achieve recommendations. UniDNR can be divided into three parts: the ID-level embedding layer, the text-level embedding layer and the rating prediction layer. Specifically, the ID-level embedding layer captures user and item embeddings based on the user-item interaction graph, which could preserve the higher-order interactive relationship by aggregating neighbor information. The text-level embedding layer embeds item descriptions and user reviews by aspect-based learning, which considering different aspects mentioned in descriptions and reviews. After learning the latent embeddings, the rating prediction layer predicts the rating that a user is most likely to give to a particular item.

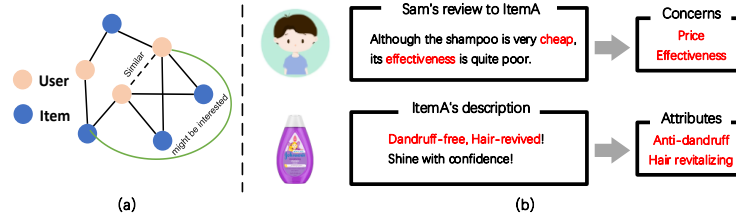


Fig. 1. (a) denotes review-based recommendation can be naturally formed as a user-item interaction graph. (b) denotes that description typically includes the inherent attributes of items which complement user reviews.

In general, we summarize our main contributions as follows. We have designed a novel framework UniDNR that learns representations from two perspectives: ID-level embedding and text-level embedding. On one hand, UniDNR captures higher-order user-item interaction by aggregating neighbor information, enabling a deeper understanding of relationships among users and items. On the other hand, UniDNR takes into account various aspects mentioned in reviews and descriptions to model users and items. By this way, UniDNR cleverly integrates reviews and descriptions to showcase their synergistic advantages. Experiments on three real-world datasets within

Amazon demonstrate the superiority of the UniDNR model compared to the state-of-the-art baselines.

The structure of this paper is described as follows. In Sect.2, we introduced the related work. In Sect.3, we detail the specific components and elaboration of our proposed model. In Sect.4, we introduced the experimental details, including experimental setup, experimental results, and so on. Finally, we draw our conclusions in Sect.5.

2 Related work

In recent years, many researchers pay much attention to improve recommender systems by leveraging user reviews[6, 8, 9, 12]. Some researchers used convolutional neural networks (CNN) to extract features from texts, to some extent improving the effectiveness of recommender systems[9, 12, 13]. For example, Zheng et al.[9] employed two parallel CNNs to capture user and item features by leveraging their reviews respectively, then predicted the ratings by FM. Building upon this foundation, Seo et al.[12] introduced two word-level attention mechanisms: the local attention and the global attention, both of which mechanisms aimed to identify more crucial words by assigning different weights to different words. Yang et al.[13] proposed an intention representation method RMCL based on mixed Gaussian distribution hypothesis which established a fine-grained connection between user reviews and item reviews. But these methods overlook the higher-order interaction relationships in the user-item graph. Some recent works designed graph neural networks to take advantage of reviews to improve model performance[14-16]. For example, RGCL[17] constructed a review-aware graph learning module to integrate reviews into graph learning. In contrast to our work, these studies model user reviews as edge relationships, and overlook the various aspects mentioned in user reviews and item descriptions.

3 The Proposed Model

In this part, we describe our proposed UniDNR model in details. The target of the recommendation task is to estimate ratings r_{ui} for any unseen user-item pair, namely predicting the rating between user u and item i which i has not interacted with u . Figure 2 shows the overall architecture of the UniDNR model.

The main idea of ID-level embedding layer is to aggregate information from neighboring nodes in the user-item interaction graph, capturing the high-order interaction relationships to generate representations for nodes.

3.1 ID-level Embedding Layer

Initializing all the inputs. The initial ID embeddings of user u and item i are initialized as one-hot encoding format, represented with \mathbf{p}_u and \mathbf{q}_i . Then \mathbf{p}_u and \mathbf{q}_i are mapped into low-dimension vectors through transform matrix $\mathbf{E}_u = \mathbf{R}^{N \times d}$ and

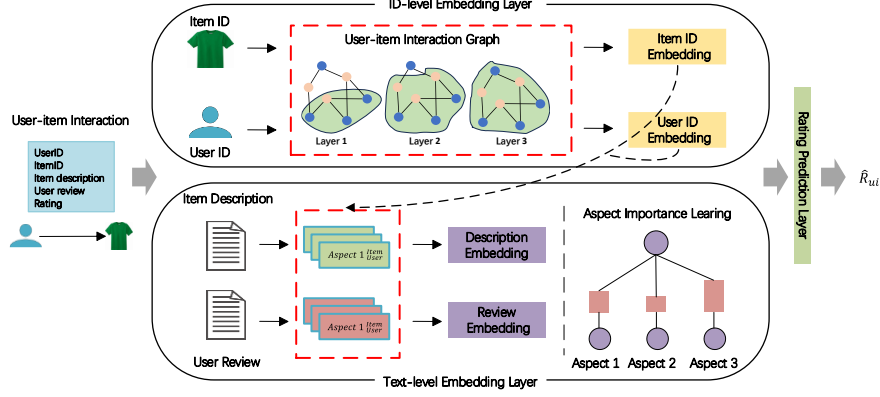


Fig. 2. The architecture of UniDNR.

$E_i = \mathbf{R}^{M \times d}$, where $\mathbf{h}_u^0 \in \mathbf{R}^d$ and $\mathbf{h}_i^0 \in \mathbf{R}^d$ denote the initial embeddings of user u and item i , and M and N denote the number of items and users, respectively:

$$\mathbf{h}_u^0 = \mathbf{E}_u^\top \mathbf{p}_u \quad (1)$$

$$\mathbf{h}_i^0 = \mathbf{E}_i^\top \mathbf{q}_i \quad (2)$$

Aggregating neighbor information. For each user and item node v , a set of neighboring nodes $N(v) = \{n_1, n_2, \dots, n_t\}$ is defined and t is the number of neighbors of node v . We aggregate its own feature representation with the representations of its neighboring nodes using aggregation functions:

$$\mathbf{h}_u^{(l+1)} = \text{Agg}(\mathbf{h}_s^{(l)}, \forall s \in N(u) \cup u) \quad (3)$$

$$\mathbf{h}_i^{(l+1)} = \text{Agg}(\mathbf{h}_s^{(l)}, \forall s \in N(i) \cup i) \quad (4)$$

where $\mathbf{h}_u^{(l)}$ is the representation of user u at layer l , $\mathbf{h}_i^{(l)}$ is the representation of item i at layer l , $\mathbf{h}_s^{(l)}$ is the representation of neighboring nodes of v at layer l and Agg is the aggregation function. The aggregation function can be chosen in different forms[18], such as mean aggregation or pooling. In this paper, we use the mean aggregation:

$$\text{MeanAgg}(\mathbf{h}_s^{(l)}, \forall s \in N(v)) = \frac{1}{|N(v)|} \sum_{s \in N} \mathbf{h}_s^{(l)} \quad (5)$$

Update node representations. After multiple layers of aggregation, the aggregated node representation is fused with the original node features to obtain the final node representation:

$$\mathbf{h}'_u = \text{Relu} \left(\mathbf{W}_h^U \text{Concat}(\mathbf{h}_u^{(l)}, p_u^0) + b \right) \quad (6)$$

$$\mathbf{h}'_i = \text{Relu} \left(\mathbf{W}_h^I \text{Concat}(\mathbf{h}_i^{(l)}, q_i^0) + b \right) \quad (7)$$

where Relu is the activation function, \mathbf{W}_h and b are the learned weights and biases. In this way, we obtained the final User embedding \mathbf{h}'_u and Item embedding \mathbf{h}'_i by aggregating the interaction graph.

3.2 Text-level Embedding Layer

The text-level embedding layer is to learn the embeddings for descriptions and reviews. We use the pre-trained language model Bert[4], which is a widely adopted text embedding approach at present, to generate the 768-dimensional representation \mathbf{o}_i for the description of item i . Similarly, the review that user u commented on item i can be represented as \mathbf{e}_{ui} .

As descriptions and reviews often include various aspects of the item and the user, and importance for different users and items varies in different aspects (For example, one may prioritize price while another may prioritize quality). It's beneficial to represent each description and review with multiple aspect embeddings, and the more important aspects should be assigned with higher weights.

Specifically, we decompose the initial embedding into different aspect embeddings. Assuming A is the set of aspects that $A = \{a_1, a_2, \dots, a_k\}$, and K is the hyperparameter, we use a fully connected layer to project each description and review embedding to multiple aspects:

$$\mathbf{a}_{i,k}^o = \mathbf{W}_k^o \mathbf{o}_i + b \quad (8)$$

$$\mathbf{a}_{u,i,k}^r = \mathbf{W}_k^r \mathbf{e}_{ui} + b \quad (9)$$

where $\mathbf{a}_{i,k}^o \in \mathbf{R}^d$ is the embedding of aspects k of description of item i , $\mathbf{a}_{u,i,k}^r \in \mathbf{R}^d$ is the embedding of aspect k for user u on item i and $\mathbf{W}_k \in \mathbf{R}^{d \times d}$ is the projection matrix.

We aim to obtain embedded representation of different aspects, so if different aspects contain the same information, the multiple aspects embeddings degenerate to be equivalent to an description embedding[19]. So, it's better if each aspect has its own unique information. Unlike traditional correlation coefficients such as Pearson, the Distance Correlation is capable of capturing not only linear relationships but also nonlinear relationships within the data. Its coefficient is zero if and only if when these vectors are independent. Therefore, we hope that the correlation coefficient between different aspects is as small as possible, so we formulate this as:

$$\text{Loss}_{\text{aspect}_o} = \sum_{m=1}^K \sum_{n=m+1}^K d\text{Cor}(\mathbf{a}_m^o, \mathbf{a}_n^o) \quad (10)$$

$$Loss_{aspect_r} = \sum_{m=1}^K \sum_{n=m+1}^K dCor(\mathbf{a}_m^r, \mathbf{a}_n^r) \quad (11)$$

and $dCor$ is the function of distance correlation defined as:

$$dCor(\mathbf{a}_m, \mathbf{a}_n) = \frac{dCov(\mathbf{a}_m, \mathbf{a}_n)}{\sqrt{dVar(\mathbf{a}_m) \cdot dVar(\mathbf{a}_n)}} \quad (12)$$

where $dCov(\mathbf{a}_m, \mathbf{a}_n)$ represents the covariance between aspect m and aspect n , $dVar(\mathbf{a}_m)$ and $dVar(\mathbf{a}_n)$ denote the respective variances of aspect m and aspect n . For more detailed definition, refer to prior works[20]. Then we use the fully connected layer to learn the importance:

$$w_{u,i,k}^o = Softmax(Concat(\mathbf{a}_{i,k}^o, \mathbf{h}'_i, \mathbf{h}'_j) \mathbf{W}_o^l + b) \quad (13)$$

$$w_{u,i,k}^r = Softmax(Concat(\mathbf{a}_{u,i,k}^r, \mathbf{h}'_i, \mathbf{h}'_j) \mathbf{W}_r^l + b) \quad (14)$$

where $w_{u,i,k}^o$ is the importance of aspect k in the description on item i for user u and $w_{u,i,k}^r$ is the importance of aspect k in the review for user u on item j . $\mathbf{W}_o^l \in R^{d \times 1}$ and $\mathbf{W}_r^l \in R^{d \times 1}$ are the learnable weights. $Softmax$ function has the capability to enhance the representation of crucial elements within the distribution by adjusting their proportions.

At last, the description and review embeddings can be represented as:

$$\tilde{\mathbf{o}}_i = \sum_{a \in A} w_{u,i,k}^o \mathbf{a}_{i,k}^d \quad (15)$$

$$\tilde{\mathbf{e}}_{ui} = \sum_{a \in A} w_{u,i,k}^r \mathbf{a}_{i,j,k}^r \quad (16)$$

3.3 Prediction layer

To model the interaction from users and items, we take the outputs $h'_u, h'_i, \tilde{\mathbf{e}}_{ui}$ and $\tilde{\mathbf{o}}_i$ as inputs, and leverage a fully connected layer to obtain the predicting rating R'_{ui} , which can be formulated as follows:

$$R'_{ui} = Concat(\mathbf{h}'_u, \mathbf{h}'_i, \tilde{\mathbf{e}}_{ui}, \tilde{\mathbf{o}}_i) \mathbf{W}_x + b \quad (17)$$

where \mathbf{W}_x and b are the learned weights and bias.

3.4 Model Optimization

Since the target of the UniDNR model is to predict user ratings for items, we employ Mean Squared Error (MSE) as the loss function to optimize the training process. This approach is widely adopted in current review-based recommender systems:

$$Loss_{mse} = \frac{1}{|S|} \sum_{(u,i) \in S} (R_{ui} - R'_{ui})^2 \quad (18)$$

where S signifies the set of user-item pairs in the training set, and R_{ui} represents the observed rating given by user u for item i . We optimize the recommendation tasks with the total loss:

$$Loss_{total} = Loss_{mse} + Loss_{aspect_o} + Loss_{aspect_r} \quad (19)$$

4 EXPERIMENTS

4.1 Experiment Settings

4.1.1 Datasets. We conduct experiments on three publicly available datasets (Giftcards, Beauty and Appliances) from Amazon datasets, which include different types of metadata[21]. These datasets include descriptions and reviews for various types of products. It is worth noting that item descriptions come from metadata in the Amazon dataset and user reviews come from review data in the Amazon data. We link user reviews and item descriptions through ItemID, meaning each data entry has the following format: [UserID, ItemID, Description, Review, Rating]. If the conditions are not met, we will delete this data entry. For each dataset, we randomly divide it into training set (60%), validation set (20%) and testing set (20%). Note that at least one interaction for each user or item is included in the training set. The statistics of the processed datasets are shown in Table 1.

Table 1. Statistics of the datasets after preprocessing.

| Dataset | items | users | entries |
|------------|-------|--------|---------|
| Giftcards | 726 | 34954 | 39106 |
| Beauty | 10645 | 148477 | 183371 |
| Appliances | 19465 | 289367 | 346250 |

4.1.2 Evaluation Metric. Following previous work, we use MSE to evaluate performance, which is widely employed for rating predictions in recommender systems. For a fair comparison, we repeat each experiment five times and report average results.

4.1.3 Baselines. We compare our UniDNR with conventionally and recently published baselines, they are list as follows:

- (1) MF, which predicts user preferences for unrated items by decomposing the user-item interaction matrix into a lower-dimensional latent feature space.
- (2) NCF, which employs a neural network to utilize embeddings for both users and items.
- (3) DeepCoNN, which employed two parallel CNN to capture feature by leveraging reviews respectively.
- (4) ABR, which employs aspect-based learning to embed items and users.
- (5) NARRE, which incorporates an attention layer to capture the informative content from reviews.
- (6) RPRM, which learns the importance of review properties to capture the usefulness of reviews.

4.1.4 Implementation Details. UniDNR is implemented using PyTorch with Nvidia RTX GPU. The size of user and item embeddings is set as $d = 32$. All the trainable parameters in our model are optimized by the Adam optimizer with a batch size in a range of $\{32, 64, 128\}$ and a learning rate in a range of $\{0.01, 0.005, 0.002, 0.001\}$. The drop rate is set in a range of $\{0.2, 0.3, 0.5\}$. The hyperparameter K is set in a range of $\{2, 3, 4\}$.

4.2 Results and Analysis

4.2.1 General Performance. Table 2 shows the experimental results of UniDNR compared with six baselines (Notice that the best results and the second best are marked in bold and underlined, respectively.). Our proposed UniDNR consistently performs best and surpasses other state-of-the-art models on three datasets, proving the effectiveness of our model. Due to the absence of text and user-item interaction information in both (1)(2), the results are poor. (3)(4)(5) model both items and users using reviews, resulting in better performances than (1) and (2). This indicates that reviews can enhance the performance of the recommendation. RPRM discerns the relative significance of different review properties in capturing the overall utility of reviews, the performance is better than other baselines.

Table 2. Recommendation performance of UniDNR and six baselines on three datasets.

| Dataset | Giftcards | Beauty | Appliances |
|----------|---------------|---------------|---------------|
| MF | 0.9814 | 0.9724 | 1.5812 |
| NCF | 0.7677 | 0.8579 | 1.5421 |
| DeepConn | 0.7466 | 0.8605 | 1.3671 |
| ABR | 0.7556 | 0.8413 | 1.3615 |
| NARRE | 0.7491 | 0.8525 | 1.4687 |
| RPRM | <u>0.7267</u> | <u>0.8375</u> | <u>1.3432</u> |
| UniDNR | 0.7116 | 0.8213 | 1.2974 |

4.2.2 Ablation Study. We remove different components in UniDNR to study their effectiveness. The results are shown in Table 3. We introduce the variants and analyze their results as follows:

Table 3. Comparison of different components of UniDNR.

| Dataset | Giftcards | Beauty | Appliances |
|------------------|-----------|--------|------------|
| UniDNR (w/o IG) | 0.7472 | 0.8499 | 1.3918 |
| UniDNR (w/o DC) | 0.7289 | 0.8369 | 1.3421 |
| UniDNR (w/o AI) | 0.7359 | 0.8407 | 1.3549 |
| UniDNR (w/o Des) | 0.7891 | 0.8735 | 1.4833 |

Without IG: We remove the embedding layer based on the user-item interaction graph in Eq.(3-7) in this variant. The degradation of the performance indicates that embedding users and items using the user-item interaction graph can better understand the higher-order relationship between users and items, thereby improving the performance of the recommender system.

Without DC: We remove the correlation loss $Loss_{aspect_o}$ and $Loss_{aspect_r}$ in Eq.(10-12) in this variant, and we find the performance becomes worse. Distance correlation loss aims to ensure that different aspects contain different information. Without distance correlation loss, different aspects might be more similar, leading to a decline in the performance of the recommender system.

Without AI: We remove the aspect importance calculation in Eq.(13, 14) in this variant, the performance is getting poor. This indicates that if the personalized preferences for different attributes of users are not taken into account, it results in less accurate recommendations.

Without Des: We remove item descriptions in datasets, the performance is becoming worse. This indicates that the information included in the item description can enhance the performance of the recommendation system.

4.2.3 Parameter Study. Table 4 shows the results of aggregating neighbors at different orders, and the best results are marked in bold. Results on different datasets exhibit variations with changes in the aggregation order, therefore, different datasets may necessitate adjusting the aggregation order to strike a balance between the local and global information. This flexibility highlights the importance of selecting an appropriate aggregation order in different contexts.

Table 4. The results of aggregating neighbors at different orders.

| Number of orders | 1 | 2 | 3 |
|------------------|---------------|---------------|--------|
| Giftcards | 0.7378 | 0.7116 | 0.7213 |
| Beauty | 0.8213 | 0.8275 | 0.8269 |
| Appliances | 1.3644 | 1.2974 | 1.3148 |

5 Conclusion

In this paper, a deep neural recommendation framework UniDNR that unites item descriptions, user reviews and user-item interaction graph is proposed to enhance the performance of recommendation. Contrasting to existing models, UniDNR considers the higher-order interaction relationships in the user-item graph and different aspects in textual information. And compared to the state-of-the-art baselines, experiments demonstrated that UniDNR can improve the performance of recommender systems.

Acknowledgements. This work was supported by the National Key Research and Development Program of China under Grant 2023YFC3304104.

References

1. Smith, B., Linden, G.: Two decades of recommender systems at Amazon. com. *Ieee internet computing* 21, 12-18 (2017)
2. Shapira, B., Rokach, L., Freilikhman, S.: Facebook single and cross domain data for recommendation systems. *User Modeling and User-Adapted Interaction* 23, 211-247 (2013)
3. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A.: Language models are few-shot learners. *Advances in neural information processing systems* 33, 1877-1901 (2020)
4. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018)
5. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013)
6. Chin, J.Y., Zhao, K., Joty, S., Cong, G.: ANR: Aspect-based neural recommender. In: *Proceedings of the 27th ACM International conference on information and knowledge management*, pp. 147-156. (2018)
7. Luo, S., Lu, X., Wu, J., Yuan, J.: Aware neural recommendation with cross-modality mutual attention. In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 3293-3297. (2021)
8. Wang, X., Ounis, I., Macdonald, C.: Leveraging review properties for effective recommendation. In: *Proceedings of the Web Conference 2021*, pp. 2209-2219. (2021)
9. Zheng, L., Noroozi, V., Yu, P.S.: Joint deep modeling of users and items using reviews for recommendation. In: *Proceedings of the tenth ACM international conference on web search and data mining*, pp. 425-434. (2017)
10. Chen, C., Zhang, M., Liu, Y., Ma, S.: Neural attentional rating regression with review-level explanations. In: *Proceedings of the 2018 world wide web conference*, pp. 1583-1592. (2018)

11. Rendle, S.: Factorization machines. In: 2010 IEEE International conference on data mining, pp. 995-1000. IEEE, (2010)
12. Seo, S., Huang, J., Yang, H., Liu, Y.: Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In: Proceedings of the eleventh ACM conference on recommender systems, pp. 297-305. (2017)
13. Yang, W., Huo, T., Liu, Z., Lu, C.: Based Multi-intention Contrastive Learning for Recommendation. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2339-2343. (2023)
14. Cai, Y., Wang, Y., Wang, W., Chen, W.: RI-GCN: Review-aware interactive graph convolutional network for review-based item recommendation. In: 2022 IEEE International Conference on Big Data (Big Data), pp. 475-484. IEEE, (2022)
15. Gao, J., Lin, Y., Wang, Y., Wang, X., Yang, Z., He, Y., Chu, X.: Set-sequence-graph: A multi-view approach towards exploiting reviews for recommendation. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 395-404. (2020)
16. Tao, Z., Wei, Y., Wang, X., He, X., Huang, X., Chua, T.-S.: Mgat: Multimodal graph attention network for recommendation. *Information Processing & Management* 57, 102277 (2020)
17. Shuai, J., Zhang, K., Wu, L., Sun, P., Hong, R., Wang, M., Li, Y.: A review-aware graph contrastive learning framework for recommendation. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval, pp. 1283-1293. (2022)
18. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. *Advances in neural information processing systems* 30, (2017)
19. Cai, W., Pan, W., Mao, J., Yu, Z., Xu, C.: Aspect re-distribution for learning better item embeddings in sequential recommendation. In: Proceedings of the 16th ACM Conference on Recommender Systems, pp. 49-58. (2022)
20. Székely, G.J., Rizzo, M.L., Bakirov, N.K.: Measuring and testing dependence by correlation of distances. (2007)
21. Ni, J., Li, J., McAuley, J.: Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp. 188-197. (2019)