A Graph Neural Network-based Video Recommendation Model Combining Users' Long-term and Short-term Preference

Mildred Brook, David Rohlf, Andrea Bishop, Matt Jones,

Abstract—With the rapid development of technology and the advancement of Internet technology, various social networking platforms are gradually coming into people's view and occupying a higher and higher position. In the recommendation scenario, the user-item interaction naturally forms a bipartite heterogeneous graph structure and with the development of graph embedding and graph neural network technologies based on deep learning to process graph domain information, the combination of graph information and recommendation systems shows strong research potential and application prospects. The methodological improvement of the recommendation algorithm based on collaborative filtering takes advantage of the nature that user-items can form a bipartite graph in the recommendation scenario. The existing methods still have some shortcomings. The methods that only use weights or convolutional recurrent neural networks to implicitly model different historical behaviors lack explicit modeling of video switching relationships in serialized behaviors. The user's interest is changing all the time, so it is not possible to recommend based on the user's history, and it is necessary to consider both the long-term and short-term interest of the user according to the video content in order to achieve accurate recommendation of short videos. In this paper, we design a recommendation model based on graph neural network, which models users' long-term and short-term interests by two vector propagation methods, respectively.

Index Terms—Recommendation; GNN; Preference

1 Introduction

With the rapid development of technology and the advancement of Internet technology, various social networking platforms are gradually coming into people's view and occupying a higher and higher position. People gradually adapt to share and record their daily lives in their social accounts. Short videos have become the most popular form of media in the 21st century. Short videos are generally short, ranging from a few seconds to a few hundred seconds. Users can shoot and post-edit videos at any time via cell phones and publish them through the online platform where they are posted. For the short video itself, the content it covers is more comprehensive and intuitive than simple text, pictures and audio, bringing more information to users and having a stronger expression. At the same time, with more appropriate background music, users can go deeper into it and have a sense of immersion, giving them a strong visual impact and inner feeling. As the number of short videos continues to increase, the phenomenon of information overload occurs. For the information overload, personalized recommendation systems [1] have become the main way for users to obtain information. Recommendation systems infer users' interests by collecting their historical behaviors, and then generate recommendation lists. It is generally believed that recommendation systems became an independent research direction when the Grouplens system was proposed by the University of Minnesota in 1994 [2, 3]. Among them, recommendation systems can be divided into content-based recommendation [4], collaborative filtering-based recommendation [5], and hybrid recommendation [6] according to the different recommendation methods [7].

Deep Learning techniques have been very successful in many artificial intelligence fields in the past decade or so, and now deep learning-based approaches are also leading the recommendation system boom and becoming the best practice in many scenarios. For example, [8] applied neural networks to collaborative filtering to improve the effectiveness of implicit recommendations, and [9] introduced embedding tasks into the recommendation domain in conjunction with task scenarios to learn representations of users and products. The paper published by Google [2] showed the mechanism of deep learning-based recommendation systems working on industrial-grade Youtube, and in [9] introduced a wide -depth model that simultaneously captures the width information of explicit features and the depth information extracted by neural networks. DeepFM [10] effectively combines the features of factorization and deep neural networks in feature learning, extracting both low-order combined features and high-order combined features, and is widely used in CTR prediction.

In the recommendation scenario, the user-item interaction naturally forms a bipartite heterogeneous graph structure, [11] and with the development of graph embedding and graph neural network technologies based on deep learning to process graph domain information, the combination of graph information and recommendation systems shows strong research potential and application prospects [12]. The methodological improvement of the

*Mildred Brook is the corresponding author.

- *Mildred Brook, David Rohlf, Andrea Bishop, Matt Jones are with Cukurova University, Turkey. (e-mail: mildredsbrook.tr@hotmail.com.*

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 November 2022 doi:10.20944/preprints202211.0111.v1

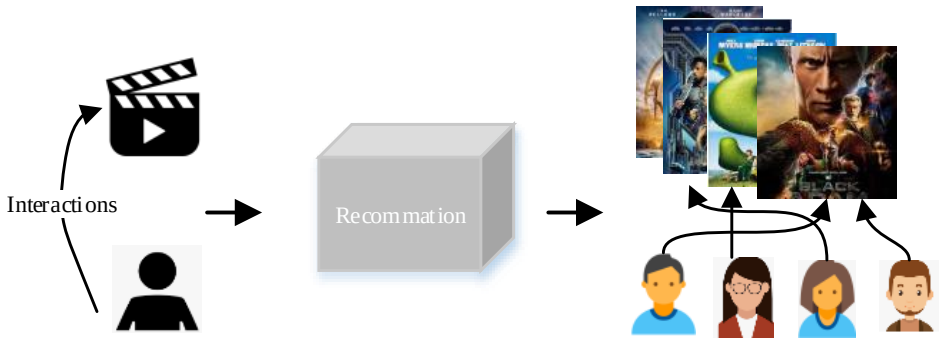


Fig. 1: Schematic diagram of recommendation system.

recommendation algorithm based on collaborative filtering takes advantage of the nature that user-items can form a bipartite graph in the recommendation scenario. The graph structure can capture multi-order neighbor relationships and multi-hop information transfer, and expand from a single interaction of ”user-item” to a higher-order interaction of ”user1-item-user2 ” to obtain higher-order historical information at that point. Based on the ability to combine with other neural network units and support end-to-end training, graph-based neural networks provide a graph perspective solution to the task of link prediction between users and items. However, one of the many current challenges is how to capture and combine the impact of other, richer, implicit user behaviors on recommendation results, such as how to incorporate implicit interaction information in graphs composed of ratings [13].

The existing methods [**?**, 14, 15] still have some shortcomings. First, the methods that only use weights or convolutional recurrent neural networks to implicitly model different historical behaviors lack explicit modeling of video switching relationships in serialized behaviors. The existing research methods are basically derived from collaborative filtering algorithms, but these algorithms use the scoring matrix as their main element, and the scoring matrix has a certain sparsity, so the accuracy of the algorithm is affected accordingly. Content-based recommendations are also plagued by the cold start problem. The current algorithms for short video recommendation are basically based on the user’s behavior or tags, and the recommendation activities are based on the user’s social attributes, while some users do not fully display their social attributes out of protection, which causes the accuracy of the recommendation is not high. Second, the current recommendation methods do not consider fine-grained modeling for users’ long-term and short-term interests. Most of the existing research methods are based on the user’s history, but since the content of short videos is richer, this recommendation method does not take into account the content of short videos, which also causes the problem of low accuracy rate. The user’s interest is changing all the time, so it is not possible to recommend based on the user’s history, and it is necessary to consider both the long-term and short-term interest of the user according to the video content in order to achieve accurate recommendation of short videos. In this paper, we design a recommendation model based on graph neural network, which models users’ long-term and short-term interests by two vector propagation methods, respectively.

2 Related Work

2.1 Recommended Methods

The recommendation system based on collaborative filtering can be divided into the following three ideas: user-based collaborative filtering method, model-based collaborative filtering method, and finally, item-based collaborative filtering method. In terms of its collaborative filtering recommendation system, the core idea is to analyze the commonality and individuality of users through their large amount of data information, and then realize the real personalized recommendation [16]. User-based system filtering approach [17], the main idea of this approach is to make use of the user ’s neighborhood combination. The so-called neighborhood set is a set in which all users have the same or similar preferences, and the users in the set are influenced by each other. Item-based collaborative filtering method [18], the main difference between this method and the above method is that this method is able to perform similarity calculation based on the user’s preference for an item. It also compares the similarity size with other items and finally recommends items in the order from largest to smallest according to the value size. Model-based collaborative filtering approach. The main idea of this approach is to use implicit features to link users with items [19]. Afterwards, the connection between the two is modeled in the form of matrix decomposition, so that user preference features can be obtained and thus personalized recommendations for users can be achieved [20]. Content-based recommendation method [21], which is one of the first and most widely used recommendation methods. This method uses the user’s personal history as a collection, calculates the characteristics of each item in the collection to form a set of user’s preference features, and then searches for items with similar preferences among all the item data to make recommendations. The key to this type of recommendation method is the topic information of the items, which is obtained from the items purchased by the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 November 2022 doi:10.20944/preprints202211.0111.v1

user, and then the topic similarity between the user and other items is calculated, and finally the top K items with the highest similarity are recommended to the user.

The so-called hybrid recommendation method [22] is to select various types of recommendation methods according to their characteristics, and form a new recommendation method through aggregation [23], weight assignment, and feature assignment [21]. In this way, the advantages of multiple recommendation methods can be integrated, which can effectively prevent the limitations of a single method. According to the actual application environment, they can be flexibly combined, and the final recommendation results obtained are more accurate [24].

2.2 Deep Learning

Deep learning techniques are widely used in the field of recommender systems, and cross-domain techniques are very common. For example, RBM restricted Boltzmann machine [25], self-encoder [26], MLP multilayer perceptron, convolutional neural network CNN [27], recurrent neural network RNN, and other neural networks. Among the above neural network structures, the network structures for user feature extraction are mainly convolutional neural networks as well as self-encoders. It is to extract the features present in a large amount of auxiliary data information. RBMs and multilayer perceptrons are mainly used to learn potential nonlinear features in the hidden factor space. Recurrent neural networks RNN and its variants [28] (gated recurrent unit GRU, long short-term memory network LSTM, etc.) are mainly used for the processing of time series.

Convolutional neural network CNN, which is one of the most widely used deep learning techniques, belongs to a kind of feed-forward neural network and also includes convolutional computation inside it, which is one of the very representative algorithms in the field of deep learning. At the same time, it also has the ability of representation learning, which can process all kinds of information input to it according to its special internal hierarchy and ensure the stability of its features.

RNNs have a stronger memory capability compared to CNNs, the various parameters input into them can be shared, and they have Turing completeness, which is not comparable to CNNs. Because of this feature, RNNs are widely used in sequence learning, and their demonstrated capability is excellent [29]. Nowadays, RNNs are mainly used in the field of speech recognition and in various types of vision problems that deal with sequence information.

2.3 Graph Neural Networks

Graph neural networks are neural networks that run directly on graph structures without converting data into sequential form, and can be divided into graph convolutional neural networks over the spectral domain represented by traditional GCNs [30, 31], and graph neural networks over the spatial domain represented by GraphSage [32]. Currently, graph neural networks have achieved excellent results in the fields of multi-document summarization, point cloud semantic segmentation, and image quizzing, etc. Meanwhile, graph neural networks make it possible to extract and apply graph structure information end-to-end by running directly on the graph structure. Since the main computations involved in GNNs include aggregating information about neighboring nodes and performing updates to the current node state, a new computational framework-message passing model is also proposed and widely used [33].

3 Methodology

3.1 Heterogeneous Construction

The goal of the video recommendation system is to satisfy the user's needs as much as possible, i.e., to recommend the videos that best match the user's preferences. In a video recommendation system, the relevant input data is the sequence of users' historical video viewing behavior, where the before and after relationships in the sequence represent the sequence of video viewing by users. The output data is a probability model that can calculate the next time a given user watches a given video. A graph is a structure with powerful data representation capabilities. In video recommendation systems, an intuitive and effective approach is to represent users and videos as two types of nodes in a graph, and to model users' viewing behavior as edges on the graph. We construct the heterogeneous graph $\mathcal{G} = [V, E]$, where V denotes the set of all nodes and E denotes the set of all edges. user nodes $u \in U$ with video nodes $i \in I$. The edges in the set E are the user-video interaction edges $r \in R$, where r_{ui} represents the existence of interaction behavior between user u and video i.

3.2 Embedding Layer

The embedding matrices P and Q are created for users and videos, respectively, where the dimension of P is $N \times D$ and the dimension of Q is $M \times D$. Where N is the number of users and M is the number of videos, D is the dimension of the low-dimensional space, which is a tunable hyperparameter with overfitting problems for too large a dimension and underfitting problems for too small a dimension. We can get the low-dimensional representation vector of each user and video by the embedding matrix of users and videos.

$$p_u = P^T v_u^U \tag{1}$$

$$q_i = Q^T v_i^I \tag{2}$$

3.3 Long-term User Preference Modeling

Our model introduces the user's history element to fuse the user's historical behavior. $[x_1, x_2, \ldots, x_m]$ denotes the item set of all short videos that users have followed, after which the same type of short videos are divided, i.e., $[h_1, h_2, \ldots, h_m]$, defined as users have followed in the history cycle of all short video types, that is, the long-term preference. Meanwhile, the newly added short video types are processed using a separate input unit as a way to ensure the stability of users' long-term preferences.

$$X = max(x_1^k, x_2^k, \ldots, x_n^k) \tag{3}$$

$$X = \frac{1}{n} \sum_{i=1}^{n} x_i^k \tag{4}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 November 2022 doi:10.20944/preprints202211.0111.v1

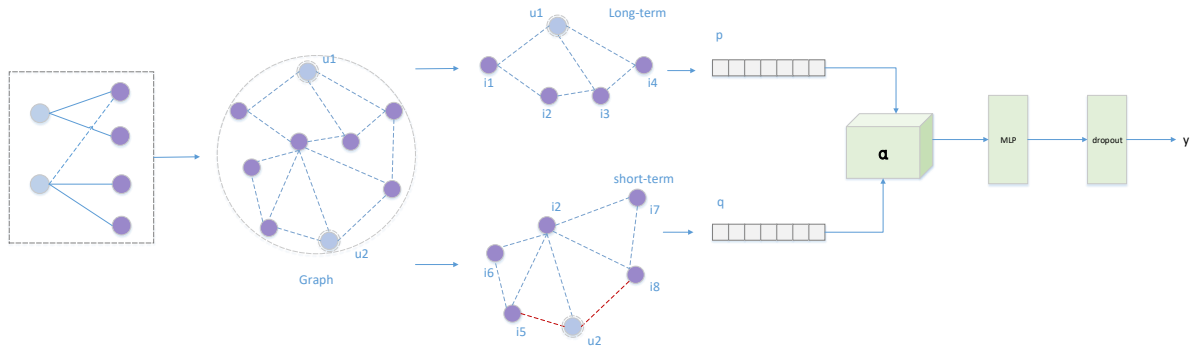


Fig. 2: Schematic diagram of our model.

where $x_1^k, x_2^k, \ldots, x_n^k$ denotes user u in the set of items of long-term preference and X denotes user in the set of long-term preference sequences.

The behavior of a user watching a video is represented as an edge between a user node and a video node on the graph. If the vector representations of all video nodes that the user has watched are propagated toward them, the aggregation naturally represents long-term preference independent of the specific time. Thus, we can model long-term preference by user node-video node vector propagation.

$$p_u^{(l+1)} = \theta(W_1^{(l+1)}(p_u^{(l)} + aggre(q_i^{(l)}|i \in R_u)) + b_1^{(l+1)}) \quad (5)$$

$$q_{i,l}^{(l+1)} = \theta(W_1^{(l+1)}(q_i^{(l)} + aggre(p_i^{(l)}|u \in R_i)) + b_1^{(l+1)}) \quad (6)$$

where θ is the nonlinear activation function, $W_1^{(l+1)}$ and $b_1^{(l+1)}$ are the network parameters and bias parameters when propagating from layer l to layer $l+1$, respectively. R_u is all videos that user u has interacted with, R_i is all users who have watched video i, and $aggre(\cdot)$ is the vector aggregation operation.

3.4 Short-term User Preference Modeling

Users' short-term preferences will be more influenced by the outside world in a short behavioral cycle, so the timeliness of short-term preferences is higher. From another perspective, it can be assumed that short-term preferences are more influential in the user's preference recommendation process, and the user's preference is likely to change at any time during a certain period of time. The above long-term preference vector propagation layer is used to portray users' long-term preferences by ignoring the historical behavioral side propagation of serial relationships. Then, we further design the vector propagation method for modeling users' short-term preferences. Considering that the user's short-term preference needs to fit with the switching behavior of video viewing, we use a vector propagation method based on the directed edges of video switching behavior.

$$q_{i,s}^{(l+1)} = \theta(W_2^{(l+1)}(q_i^{(l)} + aggre(q_j^{(l)}|j \in T_i)) + b_2^{(l+1)}) \quad (7)$$

where θ is the nonlinear activation function, $W_2^{(l+1)}$ and $b_2^{(l+1)}$ are the network parameters and bias parameters when propagating from layer l to layer $l+1$, respectively. $aggre(\cdot)$ is the vector aggregation operation.

3.5 Hybrid Recommender Framework

The outputs of the two neural network models, long-term interest of the user and short-term interest of the user, are fused in the merging layer to form a hybrid video vector output. The specific fusion method of the merging layer is shown as follows:

$$q_i^{(l+1)} = (q_{i,l}^{(l+1)}, q_{i,s}^{(l+1)}) \quad (8)$$

$$f(w_1, w_2) = \alpha w_1 + (1-\alpha) w_2 \quad (9)$$

Subsequently, we predict the viewing probability for a given user and video by means of an attention network-based prediction function.

$$\hat{y} = MLP(p + u|ATN(q_i R_u)) \quad (10)$$

where MLP represents a multilayer perceptron and ATN represents an attention network. The final output $\hat{y}$ is a probability value ranging from 0 to 1. value, the larger the value, the more likely the user u is to watch the video i.

3.6 Dropout

Neural networks are prone to overfitting in the training process, and Dropout can be applied to the access sequence, which is equivalent to the pre-processing process of removing some of the access results, and in layman's terms, it is a way to reduce the sensitivity of the model to external noise. For example, if a user watches a short video and selects a video he does not like by slip, the recommendation model constructed in this paper will not be overfitted by such external noise.

4 EXPERIMENTS

4.1 Datasets

We choose a dataset from a video website. The dataset has about 292286 pieces of data, 61030 users for 14952659 short video viewing records and user behavior.

4.2 Metrics

In order to verify the accuracy and effectiveness of the short video recommendation model proposed in this paper, the accuracy of the recommendation system is tested by various evaluation indexes, and the top K items that users like most are selected as the final recommendation results in the user feedback recommendation list.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 November 2022 doi:10.20944/preprints202211.0111.v1

TABLE 1: Statistical information of datasets.

dataset	users	items	interactions
Videos	61030	292286	14952659

TABLE 2: Model Performance Comparison on Movielens-1M.

Model	MRR	NDCG@1	NDCG@2
FM	0.823	0.779	0.892
GCN	0.848	0.822	0.906
GraphSAGE	0.873	0.836	0.913
OURS	0.892	0.844	0.935

$$Recall = \frac{recommendsitemsthatalsomeetuserrequirements}{userpreferreditems} \tag{11}$$

$$Precision = \frac{recommendsitemsthatalsomeetuserrequirement}{recommendeditems} \tag{12}$$

MRR (Mean Reciprocal Rank) evaluation index, for the Top-K recommendation list generated by the recommendation system, if a user selects the nth item in the recommendation list, then the score of the whole recommendation list is n. Then the score of the whole recommendation list is $\frac{1}{n}$. Therefore, the MRR can be calculated as follows:

$$MRR = \frac{1}{|Q|}\sum_{k=1}\frac{1}{rank_k} \tag{13}$$

where Q denotes all the items in the recommendation list, $|Q|$ denotes the length of the items in the whole recommendation list, and $rank_k$ denotes the position of the first item picked by the user in the recommendation list. In other words, if the user picks the first item in the recommendation list, the higher the accuracy rate of the recommendation system.

$$NDCG@K = \frac{1}{|U|}\sum_{u\in U}\frac{1}{(log_2(index_u+1))}, \tag{14}$$

where p_u is the corresponding true score in the test set. The higher the evaluation index score, the better the performance of the recommendation.

4.3 Baselines

- GCN. The method is a classical improvement of convolutional methods on graph networks. It is a scalable semi-supervised learning method based on graph structured data. The model varies the choice of convolution kernel by local first-order approximation of the spectral graph convolution.
- FM. The FM model takes into account the information implied by the crossover features and obtains the crossover features by inner product of two embedded features, but ignores the problem that not all feature combinations are meaningful.
- GraphSAGE. We applied GraphSAGE on the user-item graph from implicit feedback to predict the interaction between user and item.

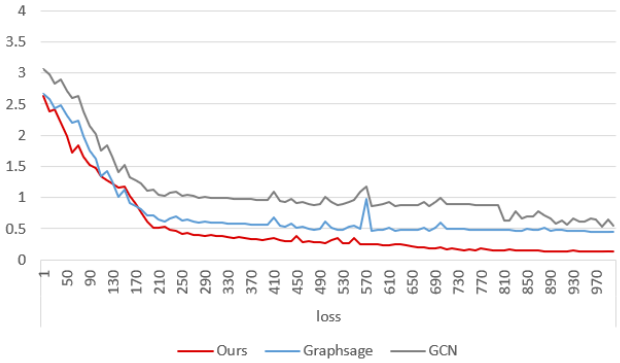


Fig. 3: Comparison of model loss curves.

4.4 Result Analysis

In this paper, different list lengths are selected for Top-K recommendation. The samples of length K are 5, 10, 15 and 20 respectively. The specific experimental results are shown in Fig. The two evaluation metrics of accuracy and recall are verified.

The results obtained from the experiments demonstrate the comparison of the video recommendation method combining the long and short-term interests of users proposed in this paper. The trends of the two evaluation indexes, recall and accuracy, are shown under the change of K value. From the figure, it can be seen that the recommendation model proposed in this paper has improved to a certain extent in both accuracy and recall rate.

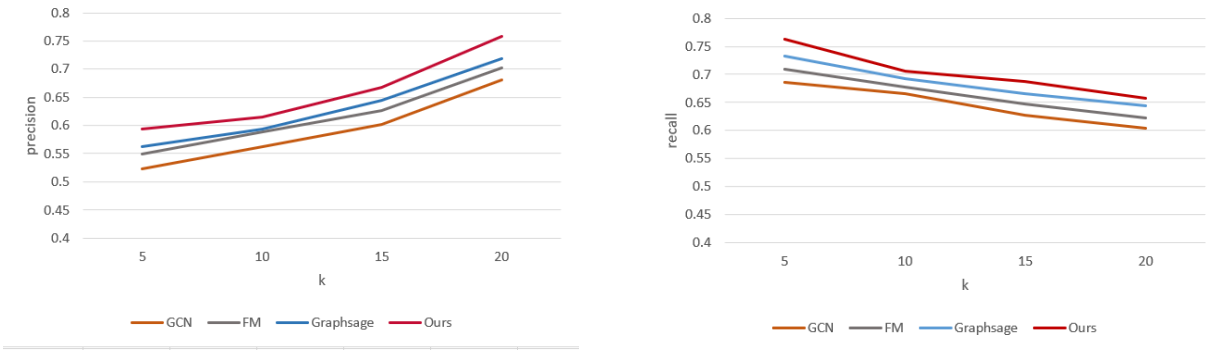
This also shows that the recommendation model proposed in this paper has certain advantages in terms of recommendation.

The α parameter serves to regulate the long-term and short-term interests of users, which will ultimately affect the output of this recommendation model. The experimental comparison of the α parameter is shown, so it can be found that the recommendation model has the highest accuracy when the α parameter is around 0.4, which can also indicate that in some perspectives, the short-term interest of the user occupies a greater influence factor on the user's choice.

5 Conclusion and Future Work

DeepLearning techniques have been very successful in many artificial intelligence fields in the past decade or so, and now deep learning-based approaches are also leading the recommendation system boom and becoming the best practice in many scenarios. The paper published by Google showed the mechanism of deep learning-based recommendation systems working on industrial-grade Youtube, and in introduced a wide -depth model that simultaneously captures the width information of explicit features and the depth information extracted by neural

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 November 2022 doi:10.20944/preprints202211.0111.v1



(a) Precision (b) Recall

Fig. 4: Expriment results comparison.

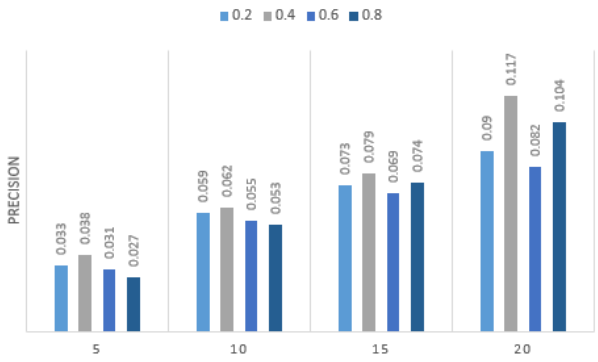


Fig. 5: α parameter.

networks. DeepFM effectively combines the features of factorization and deep neural networks in feature learning, extracting both low-order combined features and high-order combined features, and is widely used in CTR prediction.

In the recommendation scenario, the user-item interaction naturally forms a bipartite heterogeneous graph structure and with the development of graph embedding and graph neural network technologies based on deep learning to process graph domain information, the combination of graph information and recommendation systems shows strong research potential and application prospects. The methodological improvement of the recommendation algorithm based on collaborative filtering takes advantage of the nature that user-items can form a bipartite graph in the recommendation scenario. The graph structure can capture multi-order neighbor relationships and multi-hop information transfer, and expand from a single interaction of "user-item" to a higher-order interaction of "user1-item-user2 " to obtain higher-order historical information at that point. Based on the ability to combine with other neural network units and support end-to-end training, graph-based neural networks provide a graph perspective solution to the task of link prediction between users and items. However, one of the many current challenges is how to capture and combine the impact of other, richer, implicit user behaviors on recommendation results, such as how to incorporate implicit interaction information in graphs composed of ratings.

The existing methodstill have some shortcomings. First, the methods that only use weights or convolutional recurrent neural networks to implicitly model different historical behaviors lack explicit modeling of video switching relationships in serialized behaviors. The existing research methods are basically derived from collaborative filtering algorithms, but these algorithms use the scoring matrix as their main element, and the scoring matrix has a certain sparsity, so the accuracy of the algorithm is affected accordingly. Content-based recommendations are also plagued by the cold start problem. The current algorithms for short video recommendation are basically based on the user's behavior or tags, and the recommendation activities are based on the user's social attributes, while some users do not fully display their social attributes out of protection, which causes the accuracy of the recommendation is not high. Second, the current recommendation methods do not consider fine-grained modeling for users' long-term and short-term interests. Most of the existing research methods are based on the user's history, but since the content of short videos is richer, this recommendation method does not take into account the content of short videos, which also causes the problem of low accuracy rate. Secondly, the user's interest is changing all the time, so it is not possible to recommend based on the user's history, and it is necessary to consider both the long-term and short-term interest of the user according to the video content in order to achieve accurate recommendation of short videos. In this paper, we design a recommendation model based on graph neural network, which models users' long-term and short-term interests by two vector propagation methods, respectively.

6 Conflict of Interest Statement

All authors have no conflict and declare that: (i) no support, financial or otherwise, has been received from any organization that may have an interest in the submitted work ; and (ii) there are no other relationships or activities that could appear to have influenced the submitted work.

References

[1] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: a survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

 doi:10.20944/preprints202211.0111.v1

[2] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston *et al.*, "The youtube video recommendation system," in *Proceedings of the fourth ACM conference on Recommender systems*, 2010, pp. 293–296.

[3] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, pp. 175–186.

[4] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks." *Journal of Machine Learning Research*, vol. 10, no. 12, 2009.

[5] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet computing*, vol. 7, no. 1, pp. 76–80, 2003.

[6] P. Melville, R. J. Mooney, R. Nagarajan *et al.*, "Content-boosted collaborative filtering for improved recommendations," *Aaai/iaai*, vol. 23, pp. 187–192, 2002.

[7] Y. Deldjoo, M. Schedl, B. Hidasi, Y. Wei, and X. He, "Multimedia recommender systems: Algorithms and challenges," in *Recommender systems handbook*. Springer, 2022, pp. 973–1014.

[8] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173–182.

[9] M. Grbovic and H. Cheng, "Real-time personalization using embeddings for search ranking at airbnb," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 311–320.

[10] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir *et al.*, "Wide & deep learning for recommender systems," in *Proceedings of the 1st workshop on deep learning for recommender systems*, 2016, pp. 7–10.

[11] Q. Wang, Y. Wei, J. Yin, J. Wu, X. Song, and L. Nie, "Dualgnn: Dual graph neural network for multimedia recommendation," *IEEE Transactions on Multimedia*, 2021.

[12] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 1, pp. 40–51, 2006.

[13] Y. Wei, X. Wang, L. Nie, X. He, and T.-S. Chua, "Graph-refined convolutional network for multimedia recommendation with implicit feedback," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3541–3549.

[14] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 811–820.

[15] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 1059–1068.

[16] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[17] D. Zhou, B. Wang, S. M. Rahimi, and X. Wang, "A study of recommending locations on location-based social network by collaborative filtering," in *Canadian Conference on Artificial Intelligence*. Springer, 2012, pp. 255–266.

[18] S. Gong, "A collaborative filtering recommendation algorithm based on user clustering and item clustering." *J. Softw.*, vol. 5, no. 7, pp. 745–752, 2010.

[19] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[20] S. Rendle, "Factorization machines," in *2010 IEEE International conference on data mining*. IEEE, 2010, pp. 995–1000.

[21] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 426–434.

[22] M. Al-Hassan, H. Lu, and J. Lu, "A semantic enhanced hybrid recommendation approach: A case study of e-government tourism service recommendation system," *Decision Support Systems*, vol. 72, pp. 97–109, 2015.

[23] R. Burke, "Hybrid recommender systems: Survey and experiments," *User modeling and user-adapted interaction*, vol. 12, no. 4, pp. 331–370, 2002.

[24] Y. Wei, X. Wang, W. Guan, L. Nie, Z. Lin, and B. Chen, "Neural multimodal cooperative learning toward micro-video understanding," *IEEE Transactions on Image Processing*, vol. 29, pp. 1–14, 2019.

[25] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International conference on artificial neural networks*. Springer, 2011, pp. 52–59.

[26] Y. Bengio *et al.*, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[27] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 215–223.

[28] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[29] S. Petridis, Y. Wang, P. Ma, Z. Li, and M. Pantic, "End-to-end visual speech recognition for small-scale datasets," *Pattern Recognition Letters*, vol. 131, pp. 421–427, 2020.

[30] Z. Huang, Z. Wang, W. Hu, C.-W. Lin, and S. Satoh, "Dot-gnn: Domain-transferred graph neural network for group re-identification," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1888–1896.

[31] Y. Wei, X. Wang, Q. Li, L. Nie, Y. Li, X. Li, and T.-S. Chua, "Contrastive learning for cold-start recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 5382–5390.

[32] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *The world wide web conference*, 2019, pp. 417–426.

[33] J. Ma, P. Cui, K. Kuang, X. Wang, and W. Zhu, "Disentangled graph convolutional networks," in *International conference on machine learning*. PMLR, 2019, pp. 4212–4221.