Joint Embedding Multiple Feature and Rule for Paper Recommendation

Wen Li[1], Yi Xie[1,2], Yuqing Sun[1,3] (✉)

[1] School of Software, Shandong University, Jinan, China

liwen_hb@163.com,heilongjiangxieyi@163.com,
sun_yuqing@sdu.edu.cn

[2] School of Computer Science and Technology,
Shandong University, Jinan, China
[3] Engineering Research Center of Digital Media Technology,
Ministry of Education, Shandong University, Jinan, China

Abstract. It is a common way to represent paper properties as a heterogeneous academic network graph, such as authorships, citations, by which the latent features of paper can be learnt. To better integrate both text and structural features, we propose the joint embedding method for paper recommendation. We adopt a pre-trained language model to learn the paper semantic features from titles, and adopt a graph convolution network to extract the structural features from the constructed academic network graph. These two embeddings are combined together through the attention mechanism as a joint one. To clarify the *real* negative samples on uncited papers, we introduce some expert rules as the selection strategy on samples in model training, which can exclude the far-unrelated negative samples and potential positive samples. User interests are modeled by their historical publications and references and thus papers are recommended according to the relatedness between user interests and paper embeddings. We conduct experiments on the ACM academic paper dataset. The results show that our model outperforms baseline methods on personalized recommendation. We also analyze the influence of model structure and parameter setting. The results show that our sample strategy effectively improves the precision of recommendation, which illustrate that the strategy enhances the quality of training data.

Keywords: Paper Recommendation · Multiple Features · Rules

1 Introduction

There are a large number of papers published every year, which means it would take a lot of time for researchers on finding papers of interest. So personalized paper recommendation is of great significance in scientific research and academic development. And it's worth studying how to accurately capture user preferences to help them find interested paper.

Existing works always introduce various attributes of users and items besides rating matrix to build recommendation models. Since single feature cannot fully reflect user interests, it's necessary to joint these different features. In addition, in order to obtain the training data for paper recommendation, existing works often use the citation relationships between papers to construct positive and negative samples. For example, a user-paper pair with citation relationship is labeled positive, indicating that the user is interested in this paper, otherwise negative [1,2]. In fact, when users look for interested papers, not only the citation relationship is considered, but also the paper classification, keywords, and so on. Therefore, we believe sampling based on a single perspective is limited.

To tackle the above challenges, we propose the joint multi-feature and rules paper embedding model for paper recommendation (JMPR). In order to capture user interests more comprehensively, we integrate both the semantic and structural features, corresponding to paper titles and academic network graph, respectively. And we introduce rules as the sample strategy on samples in the training process. The contribution of this paper resides on two aspects:

(1) We propose the joint embedding method for paper recommendation. We construct the academic network graph based on the academic paper dataset, in which the entities such as papers, authors, venues are as nodes and the relations between them are as edges. To mine the potential relationships between entities which are not directly connected and get the paper entity representations incorporating neighborhood information and user interests, we optimize a graph convolution network (GCN) to extract the structural features from the academic network graph. To get the representations of paper title incorporating with domain knowledge, we pre-train a language model on domain dataset to learn the semantic features from titles. Then, to get the joint paper embedding, we combine the above two representations through attention mechanism. Finally, the recommendation is based on the relatedness between user interests and paper embeddings. We calculate the similarity between user vector and paper vector as the probability that the user is interested in the paper.

(2) We propose the rule-based sample selection strategy to clarify the real negative samples on uncited papers, so as to exclude the far-unrelated negative samples and potential positive samples from being mistakenly selected as negative samples. To jointly model the correlation between users and papers, we define three rules based on paper classification, references, and keywords. And according to the rules, we select the positive and negative samples as training set.

We conduct experiments on the ACM academic dataset, and compare our method with the baselines. The results show our method outperforms others on the recommendation task. Then, we analyze the influence of model structure and parameter setting. The experimental results show that our sample strategy effectively improves the precision of recommendation, which illustrate that the strategy enhances the quality of training data.

The rest of this paper is organized as follows. In section 2, we will introduce the related works. In section 3, we will introduce our proposed method in detail. In section 4, we will analyze the experiments. Finally in section 5, we will make the conclusions.

2 Related Work

2.1 Recommendations Based on Content and Rating Features

Traditional recommendation methods include content-based and CF-based methods. Content-based methods usually make recommendations by discovering the relation between user profiles and paper features. The user profiles are mainly constructed by the interaction between users and papers. For example, Gautam and Kumar [3] proposed a tag-based method that uses paper tags which users are interested in to represent user profiles. In addition, there are many ways to represent paper features. For example, Jeong et al. use BERT [4] to obtain the paper sentence representations [5]. Tao et al. uses LDA topic model [6] to obtain the paper feature representations. The basic idea of CF-based methods is that similar users' favorite items are also similar. Compared with content-based methods, CF-based methods, such as SVD [7], are more independent of the content of items, making it applicable to a wider range of scenarios. However, they rely on the rating matrix, so suffer from data sparsity and cold start problems. Through Combining the CF-based and content-based methods, the above two problems can be solved to a certain extent. For example, Sugiyama et al. use content-based method to model the user preferences, then use CF-based method to discover papers that users are potentially interested in [8]. The data feature that recommendation models mentioned above rely on is relatively single, so that it's difficult to mine potential user preferences.

2.2 Recommendations Based on Structural Features

With the development of graph-based information retrieval and data mining technology, more and more graph-based recommendation models have been proposed. For example, the emergence of social network contributed to the research of trust-aware recommendation systems [9], which could help us to infer the user's preferences indirectly by summarizing the user's friends' preferences. In addition, the Knowledge Graph (KG) as the side information besides rating matrix is also increasingly used for recommendation, such as DKN [10], PER [11], RippleNet [12], KGCN [13], KGCN-LS [14], etc. KG could improve the precision, diversity, and interpretability of the recommendation system [15]. Among the above-mentioned KG-based methods, KGCN [13] performs GCN to mine the high-order hidden information on the KG, which proves work well. And our work in this paper is inspired by this work.

There is a natural network structure between academic papers, since they are not isolated but connected with each other by citation relations, co-authors and so on. In fact, many graph-based recommendation models for paper recommendation have been proposed. For example, Pan et al. [16] proposed a model based on the similarity learning of citation network and keyword network. Manju et al. [17] proposed a model based on social network. It is also possible to combine graph-based methods with traditional methods. For example, Kong et al. [18] proposed a method that combines graph-based and CF-based methods. They use Word2vec and Struc2vec to construct citation network with semantic information, then calculate the cosine similarity between user

representation and paper representation. Since graphs are rich in information, it's necessary to find an appropriate way to make full use of them.

3 Joint Multi-feature and Rules Paper Embedding Method

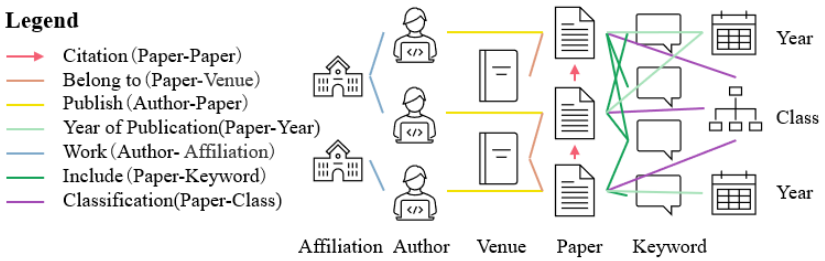


Fig. 1. Academic network graph based on ACM academic dataset.

3.1 Problem Definition

In this paper, the paper recommendation problem is formally defined as: Given a set of M users $U = \{u_1, u_2, \ldots, u_M\}$ and a set of N papers $V = \{v_1, v_2, \ldots, v_N\}$, we aim to learn a prediction function $\hat{y} = \mathrm{F}(u, v|\theta)$ that predicts whether user u has potential interest in paper v, $\hat{y}$ denotes the predicted probability that user u is interest in paper v, and θ denotes the parameters of function F.

In an academic dataset, each paper usually contains the title, keywords, authors, references, and some other attributes. As shown in Fig.1, in order to capture inter-paper relatedness, we construct an academic network graph $G = (E, R)$, where E denotes the entity set and R denote the relation set. The types of entity include "Affiliation" "Author" "Venue" "Paper" "Keyword" "Year" and "Class". The types of relation include "Citation" "Belong to" "Publish" "Year of Publication" "Work" "Include" and "Classification". Each triple of entity-relation-entity is represented as the form (e_1, r, e_2), where $e_1, e_2 \in E$, and $r \in R$. It is a common way to represent these attributes as a heterogeneous academic network graph, such as authorships, citations [2]. We construct the academic network graph based on the ACM academic dataset, then we choose it as the structural feature.

3.2 Overall Framework

Generally, a paper title is the high-level summarization of the paper content, which are rich in semantic information. And in academic network graph, papers are not isolated but connected with each other through different relations. The entities involved in a publication and its relations together constitute an informative network. The above features reveal different aspects of papers, so we consider both semantic features and structural features on modeling paper. We propose the joint multi-feature and rules paper embedding model (JMPR), as shown in Fig.2, which include three modules, i.e. the title

embedding module, academic network GCN module, and joint recommendation module.

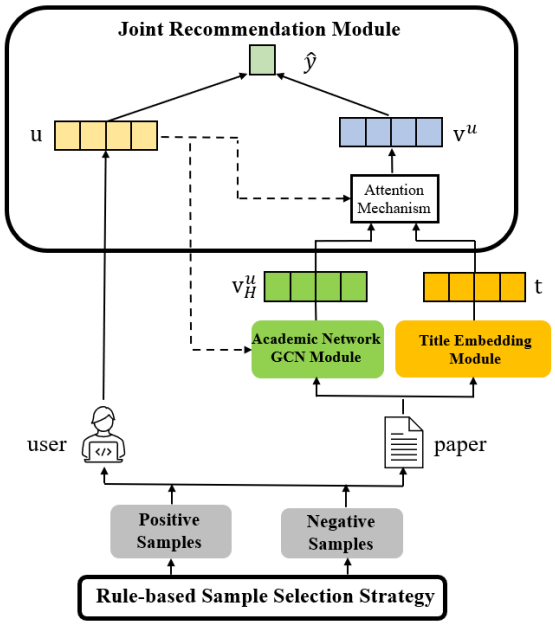


Fig. 2. JMPR model framework.

In the title embedding module, we pre-train a language model on domain dataset to get the semantic representation of paper title. In academic network GCN module, we optimize a graph convolution network (GCN) to capture the inter-paper relatedness and extract the high-order hidden information from academic network. And we obtain the representation of paper entity incorporating neighborhood information and user interests from this module. In joint recommendation module, we obtain the joint paper embedding through combining the title representation and paper entity representation mentioned above with attention mechanism. And we also obtain the user representation which incorporate the historical interest. Finally, we predict the user's potential interest in papers by calculating the vector similarity between user and paper. In addition, to handle with the sampling problem in the training process, we propose the rule-based sample selection strategy to clarify the real negative samples on uncited papers, so as to exclude the far-unrelated negative samples and potential positive samples from being mistakenly selected as negative samples.

3.3 Rule-based Sample Selection Strategy

There are two forms in citation relation: cited and uncited, which are usually directly used as positive and negative samples to train models. We call it naive sample selection strategy. However, there are cases where users and papers are far-unrelated in uncited

samples, such as a user in computer science and a paper in literary. Since the user and paper are from totally different domains, the naive sample selection strategy will lead to under-fitting problems. Any users and papers from the same domain will be predicted highly correlated by the recommendation model trained this way, which is obviously problematic. In addition, there may also be potential positive samples in uncited samples, because a user may have potential interest in the papers he didn't cite before.

To address the above problems, we propose the rule-based sample selection strategy to get user-paper pairs for training the paper recommendation model. Our strategy is user-centric. We select suitable papers for each user to construct positive samples and negative samples.

Consider user u, we take the papers that have citation relationship with u as positive samples about u and label them $y = 1$. To avoid labeling the potential positive samples and far-unrelated samples as negative samples, we introduce rules. According to the rules, we select the suitable ones from uncited papers as negative samples about u, and label them $y = 0$. By this way, the quality of training data and precision of the recommendation model are obviously improved.

Next, we will introduce three rules, and explain the sampling process in detail.

The three rules are the similarity of the research direction, Jaccard similarity of the references, and Jaccard similarity of the keywords, respectively.

Research direction. The classification systems of the academic paper research direction are usually hierarchical structures, such as the ACM Computing Classification System (CCS) in computer science. We define two papers' research direction similarity as S_1, that is, the hierarchical logarithmic distance of paper v_1 and v_2 from the bottom paper nodes to their public parent node in classification system. R_1 and R_2 denote all nodes included in the path from the bottom paper node to the root node of paper v_1 and v_2, respectively. l_i denotes the level where the current node i is located.

$$S_1(v_1,\ v_2) = -\sum_{i\in(R_1\cup R_2-R_1\cap R_2)}\frac{1}{2^{l_i}} \tag{1}$$

References. We define the two papers' Jaccard similarity of references as S_2. Ref_1 and Ref_2 denote the reference sets of paper v_1 and v_2, respectively.

$$S_2(v_1,\ v_2) = \frac{|Ref_1\cap Ref_2|}{|Ref_1\cup Ref_2|} \tag{2}$$

Keywords. We define the two papers' Jaccard similarity of keywords as S_3. K_1 and K_2 denote the keyword set of paper v_1 and v_2, respectively.

$$S_3(v_1,\ v_2) = \frac{|K_1\cap K_2|}{|K_1\cup K_2|} \tag{3}$$

We synthesize the above three rules to calculate the correlation score $S(v_1,\ v_2)$ of two papers. In fact, the above rules are always calculated 0 between most paper pairs, due to the data sparsity problem. In this paper, we believe that the correlation between different paper pairs is reflected in different aspects, and the highly relevant paper pairs may only be highly relevant in just one aspect, so we choose the maximum value of the three rules as the final correlation score of paper v_1 and v_2.

$$S(v_1, v_2) = max\left(S_1(v_1, v_2), S_2(v_1, v_2), S_3(v_1, v_2)\right) \quad (4)$$

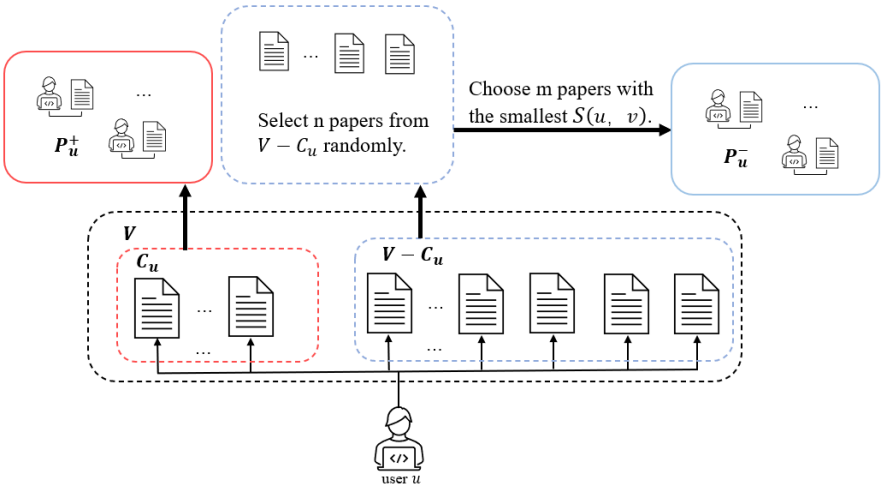


Fig. 3. Flow chart for sample selection strategy incorporating rules.

Then, we calculate the correlation score $S(u, v)$ of user u and paper v based on the correlation score $S(v_1, v_2)$ of two papers. P_u denotes the paper set published by u.

$$S(u, v) = max\{r(p_u, v) | p_u \in P_u\} \quad (5)$$

So far, we could label the positive and negative samples about user u according to $S(u, v)$. The process in detail is shown in Figure 3. V denotes the set of all papers, C_u denotes the set of papers cited by u. If u has cited paper v, then $y = 1$,indicating a positive sample, and $P_u^+ = \{v | v \in C_u\}$ denotes the set of positive samples about u. Negative samples about u are selected from the paper set $V - C_u$ that u has not cited. Since the number of samples without citation relation is huge and the calculation of $S(u, v)$ is time-consuming and space-consuming, we only randomly select n papers from $V - C_u$ to calculate $S(u, v)$, n<< $|V - C_u|$. We finally select m papers with the smallest $S(u, v)$ as the final negative sample set P_u^- about u, where m<n.

3.4 Joint Embedding based Academic Paper Recommendation

3.4.1 Title Embedding Module

Inspired by SBERT [19]'s idea of fine-tuning the model parameters of pre-trained BERT through Siamese neural network on domain dataset to output sentence embeddings, we optimize a Siamese neural network on our constructed title dataset to output title embeddings.

The title dataset is constructed based on the ACM academic paper dataset. We define z as the true correlation score between t_1 and t_2. If there is a citation relationship between (t_1, t_2), we choose it as positive sample and label it as $z = 1$. Then we select

negative samples from uncited ones based on rules. We randomly select ten times the number of positive samples from the uncited title pairs as candidate set. Then we calculate their $S(v_1, v_2)$ based on formula (4). Finally, we choose the samples with the smallest $S(v_1, v_2)$ from the candidate set as the negative samples, and we label them $\mathrm{z} = 0$. The negative sample set is the same size as the positive one.

Our Siamese neural network includes three layers, which are Bert layer, mean pooling layer and the fully connected layer in sequence. The difference from SBERT is we add a fully connected layer to reduce the output vector dimension. We use the mean square error (MSE) function as the loss function. $\hat{\mathrm{z}}$ denotes the predicted score.

$$loss = \left|\left|\mathrm{z} - \hat{\mathrm{z}}\right|\right|_2 = \left|\left|\mathrm{z} - \cos(\mathrm{t}_1, \mathrm{t}_2)\right|\right|_2 \tag{6}$$

3.4.2 Academic Network GCN module

Inspired by KGCN [13]'s idea of implementing GCN on knowledge graph, we implement GCN on academic network graph.

Consider a user-paper pair (u, v), $\mathrm{u} \in R^d$ and $\mathrm{v} \in R^d$ are vector representations of u and v. r denotes the relations between entities on the academic network graph. And $\mathrm{r} \in R^d$ denotes vector representations of r.

A user may be more interested in the papers from the same venue, another may be more interested in the papers of the same author. $r_{v,e}$ represents the relation between paper v and its neighbor entity e. We use $\phi^u_{r_{v,e}}$ to represent the score between user u and relation $r_{v,e}$. $\phi^u_{r_{v,e}}$ describes the importance of relation $r_{v,e}$ to u.

$$\phi^u_{r_{v,e}} = \mathrm{u} \times \mathrm{r}_{v,e} \tag{7}$$

We use E_v to denote the entity set directly connected to v in the academic network. In fact, the size of E_v may change a lot with different papers in the academic network graph. For computational convenience, we sample a fixed size of neighbors for each paper randomly instead of using all of them, which is defined as E'_v. $|E'_v| = K$ is a constant. $\tilde{\phi}^u_{r_{v,e}}$ is the normalized result of $\phi^u_{r_{v,e}}$.

$$\tilde{\phi}^u_{r_{v,e}} = \frac{exp(\phi^u_{r_{v,e}})}{\sum_{e' \in E'_v} exp(\phi^u_{r_{v,e'}})} \tag{8}$$

We compute the linear combination of the entities in E'_v to characterize the topological neighborhood structure of paper v. We use $\mathrm{v}^u_{E'_v}$ to denote the vector representation of paper v's neighborhood. The score $\tilde{\phi}^u_{r_{v,e}}$ between u and $r_{v,e}$ plays an important role as the personalized filter. e is the vector representation of entity e.

$$\mathrm{v}^u_{E'_v} = \sum_{e \in E'_v} \tilde{\phi}^u_{r_{v,e}} \mathrm{e} \tag{9}$$

E'_v can also be called as the single-layer receptive field of paper v. Then we aggregate paper $v's$ initial representation v and $v's$ neighborhood representation $\mathrm{v}^u_{E'_v}$ into a single vector $\mathrm{v}^u_1 \in R^d$ as $v's$ first-order representation.

$$\mathrm{v}_1^u = \sigma\left(W \cdot \left(\mathrm{v} + \mathrm{v}_{E_v'}^u\right) + b\right) \tag{10}$$

So far, we get the representation of entities in the academic network graph after single-layer GCN on academic network graph, which are also called first-order representation of entities. They only depend on itself and the neighborhood entities directly connected to them. We define the initial entity representation as zero-order representation, and the entity representation after single-layer GCN as first-order representation. In order to mine the long-distance interest of users, we extend the receptive field to multi hops which means the entities which are indirectly connected to the given entity are also selected to be its neighborhood entities. And we generate the neighborhood representation through implementing multi-layer GCN on the academic network graph. By this way, we get the high-order representations of entities. We use H to denote the maximum depth of the neighborhood. For a given user-paper pair (u, v), we compute the receptive field M of v iteratively, then generate the H-order representation $\mathrm{v}_H^u \in R^d$ of v through H times aggregation of entity representation and its neighborhood representation.

3.4.3 Joint Recommendation Module

We joint the paper title vector $\mathrm{t} \in R^m$ generated by title embedding module and paper entity vector v_H^u generated by academic network GCN module with attention mechanism. The dimension of t is bigger than v_H^u, so we add a fully connected layer to reduce the dimension of t.

$$\mathrm{t}' = \sigma(W_1 \cdot \mathrm{t} + b_1) \tag{11}$$

We use user vector u to calculate the attention weights α and β.

$$\alpha = \mathrm{u} \cdot \mathrm{t}' \tag{12}$$

$$\beta = \mathrm{u} \cdot \mathrm{v}_H^u \tag{13}$$

Then, we calculate the weighted sum of the t' and v_H^u to get the final paper representation v^u.

$$\mathrm{v}^u = \sigma(W_2 \cdot (\alpha \mathrm{t}' + \beta \mathrm{v}_H^u) + b_2) \tag{14}$$

$\hat{y}$ denotes the predicted probability that user u has potential interest in paper v, and is calculated by the inner product of paper vector v^u and user vector u.

$$\hat{y} = \mathrm{u} \times \mathrm{v}^u \tag{15}$$

We choose the cross-entropy loss function, and implement the rule-based sample selection strategy in the training process. P_u^+ and P_u^- are the positive sample set and negative sample set about user u, respectively, obtained by section 3.3. The last term is the regularization term.

$$\sum_{u \in U}\left(-\frac{1}{|P_u^+ \cup P_u^-|}\sum_{v \in P_u^+ \cup P_u^-} y log\hat{y}\right) + \lambda \|\mathrm{F}\|_2^2 \tag{16}$$

4 Experiments

4.1 Dataset

We use the ACM dataset for experiments. It contains more than 40,000 academic papers in computer science. Since these papers are domain relevant, they are suitable for verifying the effectiveness of our rule-based sample selection strategy. In addition, the dataset contains the CCS classification labels which can be used to calculate the research direction similarity defined in section 3.3. CCS refers to the ACM computing classification system, which is a standard classification system with hierarchical structure in computing science. The basic statistical information of this dataset is shown in Table 1.

Table 1. Statistics of ACM dataset

Statistics	Num
user	44953
paper	31889
sample	348856
entity	148376
relation	7
(e_1, r, e_2)	491679

4.2 Baselines and Experiment Setup

Baseline models are as follows.

SVD [7] is a traditional CF-based recommendation model, which needs rating matrix. The basic idea of SVD is to match the original data into a low-dimensional space, and calculate the predicted score of the unrated items, then recommend the items with high predicted scores to the user.

KGCN [13] implements GCN on knowledge graph. The basic idea is to aggregate the entities with their neighborhood in the knowledge graph to capture the inter-item relatedness and the potential interest of users.

KGCN-LS [14] introduces Label Propagation Algorithm (LPA) on the basis of KGCN, which is equivalent to introducing a regular term to prevent overfitting problem.

RippleNet [12] takes the items that users are interested in as seeds, and uses these seeds to spread out to other items on knowledge graph, refer to the idea of water wave propagation. This process is called preference propagation. RippleNet uses the method of spreading preferences in knowledge graph to discover the potential interests of users continuously and automatically, so that it achieves personalized recommendation.

Our model and its variants in this paper are as follows.

JMPR is the joint multi-feature and rules paper embedding model we proposed in this paper. JMPR includes three modules and implements the rule-based sample selection strategy.

ANGCN only uses academic network graph as feature, corresponding to academic network GCN module. And ANGCN implements naive sample selection strategy.

ANGCN-TE adds titles as features on the basis of ANGCN.

ANGCN-Neg replace ANGCN's naive sample selection strategy with our proposed rule-based sample selection strategy.

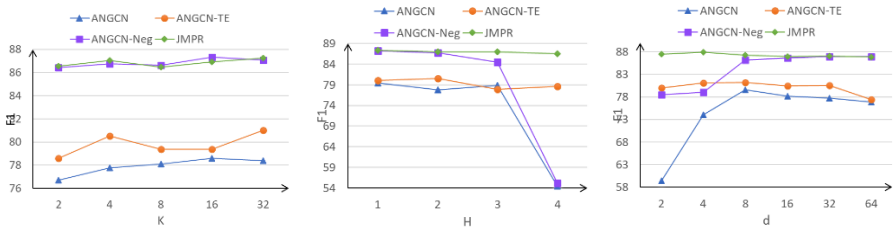
We use F1 score and AUC to evaluate the model performance on the task of judging whether users are interested in papers.

The experimental setup of JMPR is as follows. In the academic network GCN module, we use different activation functions as σ in formula (10) to aggregate the entity representation and its neighborhood representation. If it is not the last layer, we choose $ReLU$, else $tanh$. In the joint recommendation module, we feed the title embedding t to activation function $ReLU$ and $tanh$ in order to match the paper entity embedding v^u, then we choose $tanh$ as σ in formula (14) to aggregate v^u and t.

4.3 Results

Table 2. Performance comparison of different models.

MODEL	AUC	F1
SVD	50.00	66.54
KGCN	86.61	79.53
KGCN-LS	86.62	79.01
RippleNet	90.46	82.78
JMPR	**95.50**	**87.87**



(a)K (H=2, d=32) (b)H (K=4, d=32) (c)d(K=4, H=2)

Fig. 4. Performance comparison of JMPR and its variants under different experimental setting.

Performance Analysis. Firstly, we compare our model with the baselines. As shown in Table 2, our model JMPR outperforms others. The reasons lie in two aspects: One is the rule-based sample selection strategy we proposed optimize the quality of training data and improve the precision of the model. The other is JMPR joints multiple features with attention mechanism, so that it could model the user preferences more accurately. What's more, GCN works well to mine high-order hidden information in academic network graph, and pre-trained paper title embedding is meaningful and rich in semantic information, which all contribute to the improvement of the model performance.

SVD performs worst. SVD only uses the rating matrix to train the recommendation model (in this article, the rating is 1 when it's a positive sample, otherwise 0), and doesn't introduce textual features or other structural features as additional information.

As an improved model of KGCN, KGCN-LS introduces LPA on the basis of KGCN, but the performance on ACM academic paper dataset is not significantly improved compared to KGCN.

RippleNet performs better than other baselines. Since it also uses a multi-layer neighborhood structure to capture the internal associations between neighborhood entities in the academic network graph just like KGCN and JMPR. It shows the importance of neighborhood information in academic network graph for recommendation.

Next, we will analyze the impact of model structure and parameter setting on model performance. The experimental results are shown in Figure 4.

Model Structure. Title summarizes the main content of the paper, so it is rich in semantic information. Since the vocabulary of papers in different domains is quite different, the sentence embedding output by model trained in general domain is not suitable for the professional field. By pre-training the title embedding on the domain dataset, we can get the proper title representations which are meaningful and incorporated domain knowledge. In addition, the academic network graph is a heterogeneous graph which is rich in structural information. By implementing multi-layer GCN on it, we can capture the inter-paper relatedness and the long-distance interests of users. As shown in Figure 4, ANGCN-TE outperforms ANGCN, since ANGCN-TE adds title as semantic feature on the basis of ANGCN. It indicates that the title embedding module and academic network GCN module complement each other and could model user preferences in all directions. What's more, compared with the naive sample selection strategy, our proposed rule-based sample selection strategy improved the performance of the model. ANGCN-Neg and JMPR replace the naïve sample selection strategy with the rule-based one based on ANGCN and ANGCN-TE, respectively. As shown in Figure 4, ANGCN-Neg outperforms ANGCN, and JMPR outperforms ANGCN-TE. Because the rule-based strategy avoids labeling the potential positive samples and far-unrelated samples as negative samples from uncited samples. By this way, the quality of training data and the precision of the recommendation model are improved. However, it is worth mentioning that the correlation score $S(u, v)$ between user u and paper v in formula (5) is time-consuming and space-consuming, which led to a large increase in the workload.

Parameter Setting. Now we analyze the influence of neighborhood entity nodes' number K, the convolution depth H, and the embedding dimension d on the performance of the model, respectively. As shown in Figure 4, we observe the ANGCN and ANGCN-Neg are sensitive to the setting of K, H, d. Since ANGCN and ANGCN-Neg only use academic network graph as feature, and he setting of K, H, d is mainly for academic network GCN module. And when K takes the value 16, H takes 1 and 2, and d takes middle value 8, the model performance is better. Because there may be overfitting or under-fitting problem under other parameter setting conditions. Then, when the paper title feature is integrated with the academic network feature, corresponding to ANGCN-TE and JMPR, it is observed that the influence of the setting of K, H, d on the model performance is weakened. It indicates that the title embedding module makes

up the academic network GCN module's lack of information, so that it improves the robustness of the recommendation model.

5 Conclusion

In this paper, we propose the joint multi-feature and rules paper embedding model for paper recommendation. We choose the academic network graph as structural feature and paper title as semantic feature to jointly model user interests. We pre-train the title embedding on domain dataset and implement GCN on the academic network graph, then aggregate them with attention mechanism. In addition, we propose the rule-based sample selection strategy to do with the sample problem in training process. The experimental results show that our method outperform the baselines in predicting whether users are interested in the paper.

Acknowledgments. This work was supported by the Major Project of NSF Shandong Province under Grant No. ZR2018ZB0420 and the Key Research and Development Program of Shandong Province under Grant No. 2019JZZY010107.

Reference

1. Xie, Y., Wang, S., Pan, W., Tang, H., Sun, Y.: Embedding Based Personalized New Paper Recommendation. CCF Conference on Computer Supported Cooperative Work and Social Computing, pp. 558-570, Springer, Singapore((2020).
2. Zhu, Y., Lin, Q., Lu, H.: Recommending scientific paper via heterogeneous knowledge embedding based attentive recurrent neural networks. Knowledge-Based Systems 215, 106744(2021).
3. Gautam,J., Kumar, E.: An improved framework for tag-based academic information sharing and recommendation system. In: Proceedings of the World Congress on Engineering, pp.1-6., U.K (2012).
4. Devlin, J., Chang M, W., Lee ,K., Toutanova,K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv: Computation and Language(2018).
5. Jeong, C., Jang, S., Park, E., Choi,S.: A context-aware citation recommendation model with BERT and graph convolutional networks. Scientometrics 124(3), 1907-1922(2020).
6. Tao, M., Yang, X., Gu, G, Li, B.: Paper Recommend Based on LDA and PageRank. In: International Conference on Artificial Intelligence and Security, pp. 571-584, Springer, Singapore(2020).
7. Koren ,Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 426-434, USA(2008).
8. Sugiyama, K., Kan, M.Y.: Exploiting potential citation papers in scholarly paper recommendation. In: Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital libraries, pp.153–162(2013).
9. Guo, G., Zhang, J., Yorke-Smith, N.: Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems. Knowledge-Based Systems 74, 14-27(2015).
10. Wang, H., Zhang, F., Xie, X, Guo, M.: DKN: Deep knowledge-aware network for news recommendation. In: Proceedings of the 2018 world wide web conference, pp.1835-1844(2018)

11. Yu ,X., Ren, X., Sun, Y., et al.: Personalized entity recommendation: A heterogeneous information network approach. In: Proceedings of the 7th ACM international conference on Web search and data mining,pp.283-292(2014).
12. Wang, H., Zhang., F, Wang, J., et al: Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 417-426, Italy(2018).
13. Wang, H., Zhao, M., Xie, X., et al:Knowledge graph convolutional networks for recommender systems. In: Proceedings of The world wide web conference, pp. 3307-3313, USA (2019).
14. Wang, H., Zhang, F., Zhang, M., et al: Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, pp. 968-977, USA(2019).
15. Wang, X., Wang, D., Xu, C., et al: Explainable reasoning over knowledge graphs for recommendation. In: Proceedings of the AAAI Conference on Artificial Intelligence 33(01), 5329-5336, (2019).
16. Pan, L., Dai, X., Huang, S., Chen, J.: Academic paper recommendation based on heterogeneous graph. In Chinese computational linguistics and natural language processing based on naturally annotated big data , pp. 381-392, Springer, Cham(2015).
17. Manju, G., Abhinaya, P., Hemalatha, M. R., Manju, G. G.: Cold Start Problem Alleviation in a Research Paper Recommendation System Using the Random Walk Approach on a Heterogeneous User-Paper Graph. International Journal of Intelligent Information Technologies (IJIIT) 16(2), 24-48(2020).
18. Kong, X., Mao, M., Wang, W., et al. : VOPRec: Vector representation learning of papers with text information and structural identity for recommendation. IEEE Transactions on Emerging Topics in Computing(2018).
19. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. 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.3982-3992(2019).