

A paper recommendation method based on SPECTER and graph convolutional networks

Panpan Guo^{1*}, Gang Zhou¹, Jicang Lu¹, Zhufeng Li¹, Taojie Zhu¹

¹College of Data and Target Engineering, Information Engineering University, Zhengzhou, China

*Corresponding Author. Email: guopan_wpw_14517@163.com

Abstract: To address the issues of data sparsity and cold-start problems in paper recommendation, as well as the limitations of general document representation methods in representing paper datasets, this study proposes a paper recommendation method based on SPECTER and Graph Convolutional Networks (PR-SGCN method). This method leverages pre-trained document-level representation learning and the citation-aware transformer SPECTER to learn paper content representations, thereby overcoming the limitations of existing methods in representational capability. Additionally, it employs network representation learning and Graph Convolutional Networks (GCN) to mine hidden information from both attribute and structural perspectives, effectively mitigating the problems of data sparsity and cold starts. Experiments conducted on the ACL Anthology Network (AAN) and DBLP datasets demonstrate that, compared with the academic paper recommendation method based on heterogeneous graphs—Citation Recommendation based on Weighted Heterogeneous Information Network with Citation Semantic Links (WHIN-CSL)—and the academic literature recommendation method integrating network representation learning and textual information, the PR-SGCN method achieves improvements in Recall@25 by 59 and 72 percentage points, and by 67 and 65 percentage points, respectively.

Keywords: SPECTER, Graph Convolutional Networks, paper recommendation, Heterogeneous Information Network, quadruple

1. Introduction

Citation recommendation is one of the key technologies for addressing the problem of academic information overload. Its objective is to generate a reference list based on the similarity between a given manuscript or paper and candidate papers, thereby saving researchers time and effort.

Citation recommendation methods are mainly categorized into content-based methods (Content Based, CB) [1, 2], collaborative filtering methods (Collaborative Filtering, CF) [3], and graph-based methods (Graph Based, GB). Collaborative filtering methods recommend papers by leveraging explicit or implicit preferences of neighboring users [4], but they often face cold-start and data sparsity problems. Content-based methods generate recommendations by utilizing textual information, features, and related descriptions of corresponding papers and users [5], but they may also suffer from cold-start issues [6] and produce overly homogeneous recommendation results. Researchers have employed graph-based methods to overcome these issues by utilizing the relationships between nodes in the network. Traditional graph-based methods treat recommendation as a link prediction task [7]. These conventional methods often assign greater weight to classic papers with a high number of citations, whereas papers with fewer citations tend to receive less weight or are even overlooked.

In the era of big data, incorporating auxiliary information is an effective approach to alleviate data sparsity and cold-start problems. In the paper recommendation task, academic datasets contain rich nodes (such as authors, papers, publications, keywords, and research fields) and relationships (such as co-authorship, citation, publication, the inclusion relationship between papers and keywords, and the affiliation relationship between papers and research fields [8]). For example, the Heterogeneous Network-based Paper Recommendation (HNPR) method [8] constructs two types of heterogeneous information networks by utilizing co-authorship relationships, citation relationships, and the affiliation relationship between papers and research fields. The HNPR method adopts a random walk-based strategy to traverse the constructed heterogeneous information network and employs natural language methods to match word sequences for recommending papers. However, these methods struggle to handle the diversity and heterogeneity issues in citation networks.

Researchers have found that Graph Convolutional Networks (GCN) can enhance the diversity of recommendations [9]. They use GCN to learn robust latent representations of users and items in recommendation systems. The effective and efficient graph

representation learning capability of GCN has also been demonstrated in web-scale recommendation applications [10]. Some studies have applied network representation learning methods to generate paper recommendations to address the heterogeneity issues of homogeneous embedding methods. Existing research uses language models such as Doc2vec, BERT [11], and SBERT [12] to learn embeddings for nodes in the network. However, since these text representation methods are trained on general corpora, their representation capability is limited when applied to specific corpora such as academic paper databases, making it difficult to obtain accurate paper representations. In contrast, domain-specific embedding methods such as SciBERT [13] and SPECTER [14] have advantages in learning contextual node embeddings and relationships between documents. Nevertheless, network representation learning methods such as DeepWalk and Node2vec are limited in mining multiple relationships (citation, authorship, co-authorship, publication, research field, etc.) and contextual information corresponding to objects in heterogeneous information networks, making it challenging to capture researchers' preferences and generate high-quality results [15].

To address the above issues, this paper proposes a paper recommendation method based on SPECTER and Graph Convolutional Networks, referred to as the PR-SGCN method. The PR-SGCN method leverages SPECTER to learn representations of paper content to overcome the limitations of general document representation methods in academic paper datasets. It also employs network representation learning methods and Graph Convolutional Networks to learn attribute information and structural information within the network, respectively [16]. This dual approach mines more hidden information from two perspectives, helping to alleviate data sparsity and cold-start problems.

2. Related work

2.1. Graph-based paper recommendation methods

In recent years, advanced graph and network representation techniques have been employed to learn vector representations of corresponding nodes by leveraging semantic relationships between nodes within graphs or networks [17]. Various paper recommendation methods have adopted these embedding techniques to generate recommendations. Gupta et al. propose graph-based methods that integrate text-based vector representations with structural embeddings to produce recommendations [18, 19]. Reference introduces WHIN-CSL, a citation recommendation approach based on a weighted heterogeneous information network that incorporates semantic links. This method employs Node2vec to mine the semantic relationships between heterogeneous bibliographic network objects for recommendation purposes. Zhang and Ma propose the DocCit2Vec algorithm, which embeds citations into dense vectors by incorporating structural context to generate recommendations [20]. Subsequently, Graph Neural Networks (GNNs) have been utilized to capture both the semantics of neighboring nodes and relational information, thereby further improving recommendation performance. Du et al. presents an attention-based convolutional neural network approach to generate citation recommendations [21]. Ali et al. introduce GCR-GAN [22], a network embedding method that employs Generative Adversarial Networks, leveraging SPECTER [14] and denoising autoencoder networks to capture network proximity and learn semantically-preserving graph representations. Ali et al. proposes a scientific paper recommendation method [23] that utilizes SPECTER [14] and memory networks. Although these methods leverage semantic relationships to address data sparsity and the "cold-start paper" problem, they overlook the heterogeneity of the network.

In recent years, Graph Convolutional Networks (GCNs) have achieved remarkable progress in network representation learning tasks such as node classification, link prediction, and recommendation. Berg et al. [24] apply GCNs to user-item interaction bipartite graphs, transforming the matrix completion task in recommendations into a link prediction problem on graphs. Both offline and online evaluations confirm the effectiveness of GCNs in modeling user preferences. Wang et al. [25] develop a framework that utilizes GCNs for embedding propagation on a user-item unified graph, modeling higher-order connectivity. Experimental results demonstrate that leveraging GCNs for representation learning successfully improves recommendation accuracy.

2.2. Knowledge graphs

A Knowledge Graph (KG) is a directed graph composed of entities, relations, and fact triples. Formally, it is represented as $\{(h, t, r) | h, t \in E; r \in R\}$, where each triple describes a relation r from a head entity h to a tail entity t . Each triple corresponds to two nodes in a graph or network and an edge connecting them [26]. Goel et al. [27] propose the construction of Temporal Knowledge Graphs by adding timestamps to traditional knowledge graphs, representing them as quadruples $\{(h, r, t, \tau) | h, t \in E; r \in R; \tau \in \Gamma\}$, where Γ denotes a set of timestamps. Both traditional triples and temporal quadruples can only indicate the existence of a certain relationship between two entities, or that such a relationship exists within a specific time frame, but they lack descriptions regarding the strength or magnitude of the relationship.

3. Methodology

To address the issues of data sparsity and the cold-start problem, as well as the limitations of general document representation methods when applied to scholarly paper datasets, this paper proposes a paper recommendation method based on SPECTER and Graph Convolutional Networks (GCN). The overall process of the proposed method is illustrated in Figure 1. It consists of four

main steps: network construction, network merging, representation learning, and recommendation. First, SPECTER is used to learn the representations of papers. Next, a similarity function (such as the cosine similarity function) is applied to calculate the semantic similarity between any two papers, constructing a semantic network. The citation network and the paper-author network are directly extracted from the dataset. Then, these three networks are merged according to specific rules. Node2vec and GCN are used to learn attribute information and structural information within the networks, respectively. The representations learned by these two methods are concatenated to obtain the final node representations. Finally, the probabilities between nodes are calculated, and papers with higher probabilities relative to a given paper are recommended from the candidate paper set, forming a recommendation list. The following sections provide a detailed discussion of each step.

3.1. Network construction

The weighted heterogeneous information network constructed in this paper is formed by merging three networks: the semantic link network, the citation network, and the paper-author network, as shown in Figure 2. The citation network and the paper-author network are directly constructed by extracting data from the dataset. In contrast, the semantic network requires embedding the content of papers and calculating their semantic similarities before it can be constructed.

3.1.1. Paper content representation

This paper uses SPECTER to learn the representations of papers. SPECTER first encodes the text of papers using a Transformer Language Model (LM), which is defined as:

$$c_{p_i} = \text{Transformer}(\text{input})_{[CLS]} \quad (1)$$

where represents the forward function of the Transformer, and corresponds to the first token vector of the input text. SPECTER learns node embeddings by optimizing the following margin-based loss objective:

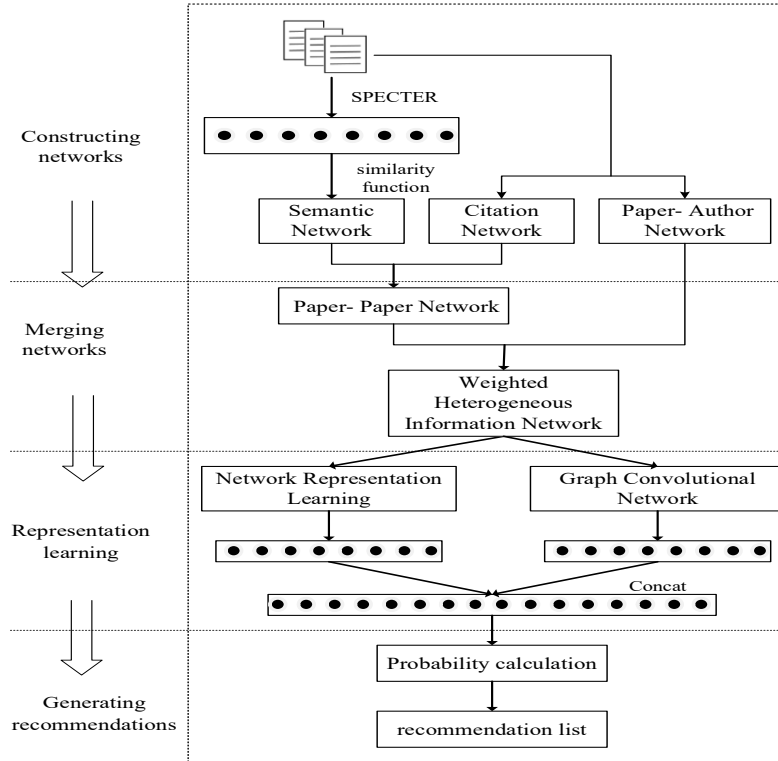


Figure 1. Process diagram of the PR-SGCN method

$$L_{loss} = \max\{d(p^q, p^+) - d(p^q, p^-) + w, 0\} \quad (2)$$

Where: p^q , p^+ and p^- and represent the query paper, the relevant paper, and the irrelevant paper, respectively. $d()$ denotes the Euclidean norm distance, and w denotes the margin distance. During training, the goal is to minimize the distance between p^+ and p^q , while maximizing the distance between p^- and p^q . During inference, the SPECTER model outputs content-based paper embeddings by activating the pooling layer of its Transformer component for the input paper.

3.1.2. Semantic network construction

A semantic network is constructed by selecting papers with high semantic similarity scores. The semantic network can be represented as $G_1 = (X_1, R_1, W_1, \phi_1, \psi_1, \theta_1)$, where $X_1 = (P)$ contains a single type of node: papers; $R_1 = (PP)$ includes a single type of relation: semantic similarity relation; and $\omega_1 = (\omega_{PP})$ is the weight of relation R_1 . $\phi_1: V_1 \rightarrow X_1$, $\psi_1: E_1 \rightarrow R_1$ and $\theta_1: W_1 \rightarrow \omega_1$ are the mapping functions for node types, relation types, and attribute types, respectively. The weight of the semantic similarity relation is defined as:

$$\omega_1 = \frac{\overline{f(p_i)} \cdot \overline{f(p_j)}}{\|f(p_i)\| \cdot \|f(p_j)\|} \quad (3)$$

Where $f(\cdot)$ represents the output of the SPECTER model.

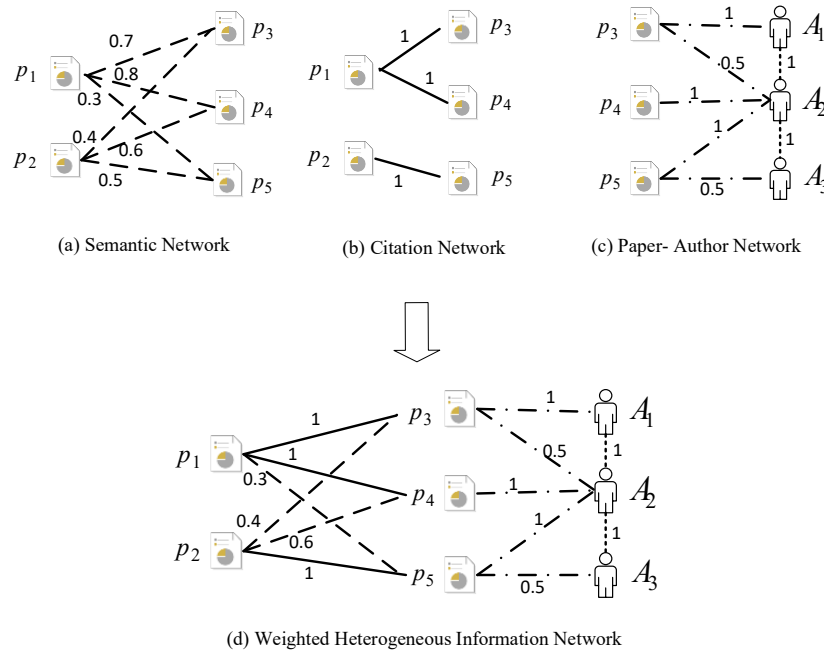


Figure 2. Formation diagram of the weighted heterogeneous information network

Note: The figures sequentially represent (a) the semantic network, (b) the citation network, (c) the paper-author network, and (d) the weighted heterogeneous information network.

3.2. Network merging

The three networks in this study are all weighted networks, which can be represented by a quadruple (h, r, t, ω_r) . Based on the original triplet, this quadruple incorporates the weight information of the relationship, allowing a more comprehensive description of the weighted network from the perspective of relational weights.

When merging networks in this study, three scenarios arise, corresponding to three merging strategies. As shown in Figure 2, given a network (such as the semantic network):

(a) If two nodes and their relationship in another network (such as the citation network) already exist in the given semantic network, but the weight value of the relationship between these two nodes differs from that in the semantic network, the larger weight value is retained and assigned to the edge. The corresponding merging rule is as follows:

$$(p_{i_1}, r_1, p_{i_2}, \omega_1) \cup (p_{i_1}, r_2, p_{i_2}, \omega_2) = (p_{i_1}, r_2, p_{i_2}, \omega_2) \quad (4)$$

In this equation, p_{i_1} and p_{i_2} represent paper nodes, r_1 denotes the semantic similarity relationship, and ω_1 represents the weight of the relationship r_1 between paper p_{i_1} and p_{i_2} . The value of ω_1 corresponds to the semantic similarity between papers p_{i_1} and p_{i_2} . Similarly, r_2 represents the citation relationship, and ω_2 denotes the weight of the relationship r_2 between papers p_{i_1} and p_{i_2} . In other words, if two papers share both a semantic similarity and a citation relationship, only the larger weight value and its corresponding relationship are retained.

(b) If two nodes in another network (such as the citation network) already exist in the given semantic network, but there is no edge between them, the semantic network remains unchanged.

(c) If one or both nodes from another network (such as the citation network) do not exist in the given semantic network, those nodes, along with their corresponding relationships and associated weight values, need to be added to the semantic network.

3.3. Representation learning

This study employs Node2vec and GCN to learn the node representations in the network. Node2vec captures the attribute information within the network, while GCN learns the structural information. The core idea of Node2vec is to perform random walks on the graph structure composed of objects. To enable more flexible sampling, both Breadth-First Search (BFS) and Depth-First Search (DFS) strategies are designed to generate a neighborhood set of nodes and are executed simultaneously during the random walk process. The optimization objective is to maximize the logarithmic probability of a node v given its neighborhood $N_s(v)$. As a result, each node v will obtain a feature representation $g_N(v)$.

The basic idea of GCN is to iteratively aggregate the neighborhood information of nodes to obtain a continuous representation of the entire graph structure. In this study, the input of GCN consists of the ID features of papers and authors, while the paper embeddings learned by SPECTER are used as the initial node features at layer 0. In each graph convolutional layer, each node propagates its current embedding to all its neighbors and aggregates the information from its neighborhood to update its embedding. For neighborhood aggregation, this study employs an average pooling method combined with a feature transformation matrix and a nonlinear activation function. Formally, the feature vector representation of node v at layer l is denoted as h_v^l , and the update rule is as follows:

$$h_{AGG}^l = MEAN(h_j^{l-1}, \forall j \in N(v)) \quad (5)$$

$$h_v^l = \tanh(W^l h_{AGG}^l) \quad (6)$$

In these equations, $N(v)$ denotes the set of sampled neighbors of node v . After the final graph convolutional layer, this study obtains the feature representation $h_G(v)$ for each node v .

Finally, the representations $g_N(v)$ and $h_G(v)$ are concatenated to obtain the final embedding representation:

$$g(v) = \text{concat}(g_N(v), h_G(v)) \quad (7)$$

3.4. Recommendation

For each target paper tp_i , the objective is to recommend a set of papers from the candidate paper set $CP = (cp_1, cp_2, \dots, cp_j)$. The probability between different papers is computed using the inner product:

$$P_r(tp_i, cp_j) = \sigma(g(tp_i), g(cp_j)) \quad (8)$$

In this equation, σ represents the activation function, and $g(tp_i)$ and $g(cp_j)$ represent the final embedding representations of the target paper tp_i and the candidate paper cp_j , respectively. Finally, the top-K papers with the highest ranking are recommended.

4. Experimental validation and analysis

4.1. Datasets and evaluation metrics

This study conducts experimental validation using papers published between 1965 and 2012 in AAN and between 1970 and 2006 in DBLP (<https://www.aminer.cn/billboard/citation>). The AAN dataset (<https://github.com/AHULiuYang/WHIN-CSL>) contains 3,936 authors and 13,275 papers, while the DBLP dataset comprises 40,126 authors and 25,056 papers. Articles published in 2012 and 2006 are respectively selected as the target papers in the AAN and DBLP datasets.

Two commonly used evaluation metrics are Recall rate (Recall) and Normalized Discounted Cumulative Gain (NDCG). The calculation formulas are as follows:

$$Recall@k = \frac{1}{Q} \sum_{j=1}^Q \left(\frac{R_p \cap T_p}{T_p} \right) \quad (9)$$

$$NDCG@k = \frac{1}{Q} \sum_{i=1}^Q \frac{2^{r_i-1}}{\log_2(i+1)} \quad (10)$$

where Q represents the number of target papers, and k denotes the number of recommended documents. Given a target paper p in the test set, R_p is the set of recommended documents, while T_p refers to the set of references cited by p . r_i is the relevance score of the i -th document.

4.2. Baseline methods

- Content-Based [28]: This method uses word2vec to learn textual information such as titles and keywords for recommendation purposes.
- Co-citation [29]: If two articles are frequently co-cited, the relationship between them is considered closer.
- Hybrid Method [30]: A simple combination of the content-based method and co-citation.
- Distributed Bag of Words of Paragraph Vector (PV-DBOW) [31]: PV-DBOW is an improvement on word2vec, which calculates similarity scores based on textual information.
- Academic Paper Recommendation Based on Heterogeneous Graphs [32]: This method employs a multi-layer graph similarity learning algorithm to calculate similarity scores between papers in the network.
- Node2vec: Scalable Feature Learning for Networks: Node2vec is a graph embedding method that integrates both DFS and BFS neighborhood sampling strategies. It can be considered an extension of DeepWalk.
- WHIN-CSL: Citation Recommendation Method for Weighted Heterogeneous Information Networks with Semantic Links [15]: WHIN-CSL is a citation recommendation approach that captures semantic relationships and attribute values on relations between vertices in weighted heterogeneous information networks.
- Academic Literature Recommendation Method Integrating Network Representation Learning and Textual Information [33]: This method generates feature vector representations based on citation networks, long text content using BERT, and short text tags, respectively. The two-stage feature fusion yields a comprehensive similarity matrix of papers, from which a recommendation list is generated.

4.3. Experimental settings

SPECTER's embedding dimension is set to 256, and the loss margin parameter $w = 1$. For training, SPECTER uses two hard negative samples and three easy negative samples. Each training batch contains 16 samples, with a learning rate of 0.001. For GCN, the number of training epochs is set to 200, with a learning rate of 0.01, using the Adam optimizer. The number of hidden layers is 2, and each hidden layer has a dimensionality of 128.

4.4. Experimental results and analysis

4.4.1. Overall performance comparison of methods

The performance comparison between the proposed method and baseline methods is presented in Table 1 and Table 2. In the tables, NDCG refers to the average of NDCG@25, NDCG@50, NDCG@75, and NDCG@100. According to the experimental results, the PR-SGCN method significantly outperforms the baseline methods on both the AAN and DBLP datasets.

Table 1. Performance comparison of methods on the AAN dataset

Method	Recall				NDCG
	25	50	75	100	
Content-Based	0.134	0.180	0.214	0.242	0.285
Co-Citation	0.143	0.178	0.202	0.216	0.302
Hybrid Method	0.147	0.196	0.233	0.257	0.306
Paragraph Vector Distributed Bag of Words (PV-DBOW)	0.178	0.237	0.280	0.311	0.289
Academic Paper Recommendation Based on Heterogeneous Graph	0.207	0.276	0.338	0.378	0.350
Citation Recommendation Method Based on Weighted Heterogeneous Information Network with Semantic Links (WHIN-CSL)	0.302	0.402	0.467	0.516	0.407
Scalable Feature Learning Method for Networks (Node2vec)	0.844	0.915	0.945	0.962	0.832
Academic Literature Recommendation Method Integrating Network Representation Learning and Text Information	0.222	0.294	0.355	0.387	0.361
Proposed Method	0.898	0.955	0.976	0.985	0.925

Table 2. Performance comparison of methods on the DBLP dataset

Method	Recall				NDCG
	25	50	75	100	
Citation Recommendation Method Based on Weighted Heterogeneous Information Network with Semantic Links (WHIN-CSL)	0.385	0.487	0.544	0.586	0.418
Scalable Feature Learning Method for Networks (Node2vec)	0.883	0.928	0.942	0.950	0.889
Academic Literature Recommendation Method Integrating Network Representation Learning and Text Information	0.259	0.348	0.410	0.457	0.383
Proposed Method	0.911	0.944	0.954	0.958	0.912

From the results in Table 1 and Table 2, it can be observed that, first, simple methods (such as content-based, co-citation, hybrid methods, and PV-DBOW) perform poorly. Single pieces of information are insufficient to capture the true differences between papers.

Second, the use of auxiliary information has improved the performance of the heterogeneous graph-based academic paper recommendation method and the hybrid network representation learning with text information method to some extent. However, these two methods are still limited by insufficient auxiliary information, and the degree of performance improvement due to auxiliary information is limited.

Finally, due to the use of more complex network methods and the incorporation of richer auxiliary information, the WHIN-CSL, Node2vec, and the proposed PR-SGCN method have achieved relatively better results. Node2vec is the second-best performing method, only following the PR-SGCN method. The comparative experimental results show that, on the AAN and DBLP datasets, the Recall@25 of the PR-SGCN method is improved by 197.35% and 136.62%, respectively, compared to WHIN-CSL, and by 6.40% and 3.17%, respectively, compared to Node2vec.

These three methods all leverage co-authorship relationships, writing relationships, citation relationships, and semantic similarity relationships. However, different information processing methods have varying impacts on different recommendation approaches. The PR-SGCN method not only uses Node2vec to capture the attribute information of the network but also uses GCN to capture the structural information of the network. In the comparative method, Node2vec represents vertices on the network constructed by the PR-SGCN method. The performance of Node2vec is second to that of PR-SGCN because Node2vec does not utilize the structural information of the network. There are two reasons why the PR-SGCN method outperforms WHIN-CSL: First, the PR-SGCN method uses SPECTER to learn semantic-aware representations of papers, while WHIN-CSL uses doc2vec to learn text representations. Since doc2vec struggles to capture citation information between papers, the learned paper representations are not accurate enough. Second, the PR-SGCN method processes citation relationships and semantic similarity relationships separately, preventing the loss of information. WHIN-CSL, on the other hand, does not process citation relationships and semantic similarity relationships separately when constructing the network, leading to the loss of some citation relationships and a lower-quality constructed network.

4.4.2. Parameter sensitivity analysis

From Figures 3 and 4, it can be seen that, on the AAN and DBLP datasets, the PR-SGCN method performs best when the number of links is 5 for both datasets. This paper suggests that, at this point, the PR-SGCN method captures a balanced state between attribute and structural information. When the number of links is smaller than the optimal number (e.g., when the number of links is 1), the performance of the PR-SGCN method is not significantly affected because, during the representation learning process, the citation and structural information compensate for the lack of content information to some extent. When the number of links exceeds the optimal number, the performance of the PR-SGCN method decreases as the number of links increases. When the number of links becomes too large (e.g., when the number of links is 100), the introduction of too much content information interferes with the network's attribute and structural information, leading to the recommendation of too many papers biased towards content similarity, significantly impacting the performance.

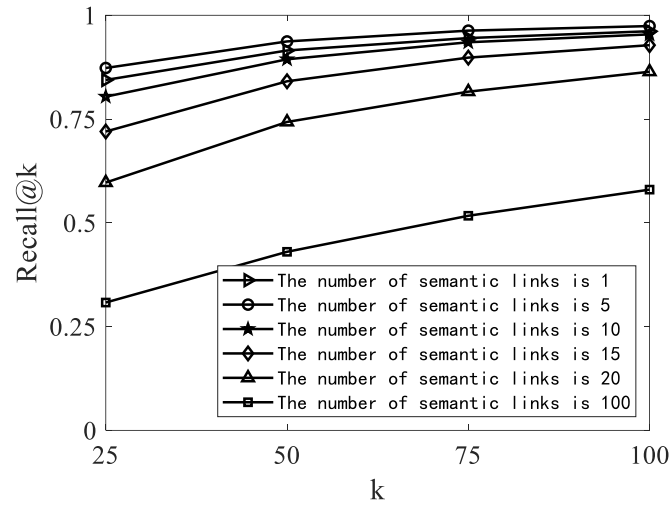


Figure 3. The impact of semantic link count on recall of the method on the AAN dataset

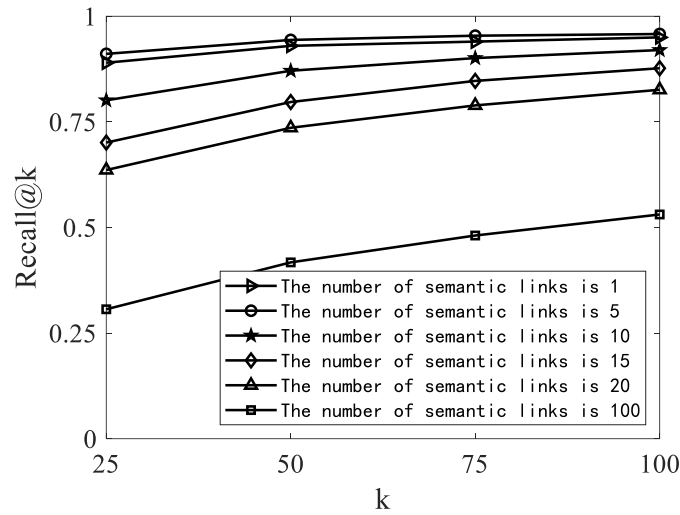


Figure 4. The impact of semantic link count on recall of the method on the DBLP dataset

4.4.3. Ablation experiment

This paper analyzes the impact of citation relationships and semantic similarity relationships on the performance of the proposed method from the perspectives of Recall and NDCG on both datasets. The experimental results comparing different variants of the proposed method are shown in Tables 3 and 4.

The results indicate that, compared to other variants, the method with the removal of the semantic network performs the worst in recommending the top 25, 50, 75, and 100 papers, followed by the removal of the citation network. This suggests that the impact of semantic similarity relationships on the results is greater than that of citation relationships.

Table 3. Performance comparison of methods on the AAN dataset

Method	Recall				NDCG
	25	50	75	100	
- Semantic Similarity Network	0.874	0.934	0.961	0.975	0.921
- Citation Network	0.868	0.934	0.961	0.974	0.918
PR-SGCN Method	0.898	0.955	0.976	0.985	0.925

Table 4. Performance comparison of methods on the DBLP dataset

Method	Recall				NDCG
	25	50	75	100	
- Semantic Similarity Network	0.887	0.927	0.934	0.950	0.893
- Citation Network	0.898	0.934	0.947	0.953	0.863
PR-SGCN Method	0.911	0.944	0.954	0.958	0.912

5. Conclusion

In response to the issues of data sparsity and cold start in paper recommendation, this paper proposes a paper recommendation method based on SPECTER and Graph Convolutional Networks. The PR-SGCN method uses network representation learning and graph convolutional networks to learn the attribute and structural information in the network, respectively, and extracts more hidden information from both aspects, significantly alleviating the sparsity and cold start problems. This paper also verifies the importance of co-authorship and citation relationships to the proposed method. When utilizing auxiliary information, it is necessary to capture researchers' preferences from different angles in order to generate more personalized recommendations, ultimately improving the performance of the method. The information utilized in this paper is limited, and there is more information available in reality, such as author affiliation, keywords, research fields, publication journals, and publication time. Future work will focus on these aspects to further improve the recommendation performance of the method.

References

- [1] Abbas, M. A., Ajayi, S., Bilal, M., Oyegoke, A., Pasha, M., & Ali, H. T. (2024). A deep learning approach for context-aware citation recommendation using rhetorical zone classification and similarity to overcome cold-start problem. *Journal of Ambient Intelligence and Humanized Computing*, 15(1), 419-433.
- [2] İlyas, K., Sinem, B., & Savas, O. (2023). A Novel Article Recommendation System Empowered by the Hybrid Combinations of Content-Based State-of-the-Art Methods. *International Journal of Applied Mathematics Electronics and Computers*, 11(1), 1-12.
- [3] Wang, H., & Li, W. J. (2014). Relational collaborative topic regression for recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 27(5), 1343-1355.
- [4] Wang, C., & Blei, D. M. (2011). Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 448-456). <https://doi.org/10.1145/2020408.2020480>
- [5] Ali, Z., Ullah, I., Khan, A., Jan, A. U., & Muhammad, K. (2021). An overview and evaluation of citation recommendation models. *Scientometrics*, 126(5), 4083-4119. <https://doi.org/10.1007/s11192-021-03909-y>
- [6] Dong, Y. F., Qu, X. Q., Li, L. H., & Dong, Y. (2022). An academic journal recommendation algorithm based on author preferences. *Computer Applications*, 42(1), 50-56.
- [7] Fu, H. J., Xiong, Y., & Zhu, Y. Y. (2020). TLP: A temporal link prediction algorithm in dynamic networks. *Computer Engineering*, 46(1), 67-73.
- [8] Du, N., Guo, J., Wu, C. Q., Hou, A., Zhao, Z., & Gan, D. (2020). Recommendation of academic papers based on heterogeneous information networks. In *2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA)* (pp. 1-6). IEEE. <https://doi.org/10.1109/AICCSA50499.2020.9316516>
- [9] Zheng, Y., Gao, C., He, X., Li, Y., & Jin, D. (2020). Price-aware recommendation with graph convolutional networks. arXiv. <https://doi.org/10.48550/arXiv.2003.03975>
- [10] Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., & Leskovec, J. (2018). Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)* (pp. 974-983). Association for Computing Machinery. <https://doi.org/10.1145/3219819.3219890>
- [11] Jeong, C., Jang, S., Shin, H., Park, E., & Choi, S. (2020). A context-aware citation recommendation model with BERT and graph convolutional networks. *Scientometrics*, 124(3), 1907-1922. <https://doi.org/10.1007/s11192-020-03561-y>
- [12] Reimers, N., & Gurevych, I. (2019). 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). Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1410>
- [13] Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *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. 3615-3620). Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1371>
- [14] Cohan, A., Feldman, S., Beltagy, I., Downey, D., & Weld, D. S. (2020). SPECTER: Document-level Representation Learning using Citation-informed Transformers. arXiv:2004.07180 [cs.CL]. <https://doi.org/10.48550/arXiv.2004.07180>
- [15] Chen, J., Liu, Y., Zhao, S., & Zhang, Y. (2019). Citation Recommendation Based on Weighted Heterogeneous Information Network Containing Semantic Linking. In *2019 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 31-36). Shanghai, China. <https://doi.org/10.1109/ICME.2019.00014>
- [16] Gao, H. Y., Zhang, X. H., & Wang, J. (2021). Link prediction algorithm based on generative adversarial hierarchical network representation learning. *Computer Engineering*, 47(2), 60-68, 76.

- [17] Liu, F., Wang, B., Zou, R., & Zhao, H. (2021). A recommendation algorithm for network representation learning based on random walks. *Computer Engineering*, 47(9), 90-96, 105.
- [18] Gupta, S., & Varma, V. (2017). Scientific article recommendation by using distributed representations of text and graph. In *Proceedings of the 26th international conference on world wide web companion (WWW '17 companion)* (pp. 1267–1268). International World Wide Web Conferences Steering Committee. <https://doi.org/10.1145/3041021.3053062>
- [19] Kong, X., Mao, M., Wang, W., Liu, J., & Xu, B. (2018). VOPRec: Vector representation learning of papers with text information and structural identity for recommendation. *IEEE Transactions on Emerging Topics in Computing*, 9(1), 226-237. <https://doi.org/10.1109/TETC.2018.2830698>
- [20] Zhang, Y., & Ma, Q. (2020). Citation recommendations considering content and structural context embedding. In *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)* (pp. 1–7). IEEE. <https://doi.org/10.1109/BigComp48618.2020.0-109>
- [21] Du, Z., Tang, J., & Ding, Y. (2019). POLAR: Attention-based CNN for one-shot personalized article recommendation. In M. Berlingerio, F. Bonchi, T. Gärtner, N. Hurley, & G. Ifrim (Eds.), *Machine learning and knowledge discovery in databases: ECML PKDD 2018* (Lecture Notes in Computer Science, Vol. 11052, pp. 675–690). Springer. https://doi.org/10.1007/978-3-030-10928-8_40
- [22] Ali, Z., Qi, G., Muhammad, K., Kefalas, P., & Khuro, S. (2021). Global citation recommendation employing generative adversarial network. *Expert Systems with Applications*, 180, 114888. <https://doi.org/10.1016/j.eswa.2021.114888>
- [23] Ali, Z., Qi, G., Kefalas, P., Khuro, S., Khan, I., & Muhammad, K. (2022). SPR-SMN: Scientific paper recommendation employing SPECTER with memory network. *Scientometrics*, 127(11), 6763-6785. <https://doi.org/10.1007/s1192-022-04425-3>
- [24] Van den Berg, R., Kipf, T. N., & Welling, M. (2017). *Graph convolutional matrix completion*. arXiv:1706.02263. <https://doi.org/10.48550/arXiv.1706.02263>
- [25] Wang, X., He, X., Wang, M., Feng, F., & Chua, T.-S. (2019). Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019)* (pp. 165–174). ACM. <https://doi.org/10.1145/3240508.3240556>
- [26] Chen, W., Wen, Y., Zhang, X., Yang, N., & Zhao, S. (2020). An improved TransE-based knowledge graph representation method. *Computer Engineering*, 46(5), 63-69, 77.
- [27] Goel, R., Kazemi, S. M., Brubaker, M., & Poupart, P. (2019). *Diachronic embedding for temporal knowledge graph completion*. arXiv:1907.03143. <https://doi.org/10.48550/arXiv.1907.03143>
- [28] Kazemi, B., & Abhari, A. (2017). A comparative study on content-based paper-to-paper recommendation approaches in scientific literature. In *Proceedings of the 20th Communications & Networking Symposium (CNS '17)* (pp. 1–10). Society for Computer Simulation International. <https://doi.org/10.5555/3107979.3107984>
- [29] Small, H. (1973). Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science*, 24(4), 265-269. <https://doi.org/10.1002/asi.4630240406>
- [30] Kanakia, A., Shen, Z., Eide, D., & Wang, K. (2019). A scalable hybrid research paper recommender system for Microsoft Academic. In *The World Wide Web Conference (WWW '19)* (pp. 2893–2899). Association for Computing Machinery. <https://doi.org/10.1145/3308558.3313700>
- [31] Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML'14)* (Vol. 32, pp. 1188–1196). JMLR.org. <https://doi.org/10.5555/3044805.3045025>
- [32] Pan, L., Dai, X., Huang, S., & Chen, J. (2015). Academic paper recommendation based on heterogeneous graph. In M. Sun, Z. Liu, M. Zhang, & Y. Liu (Eds.), *Chinese computational linguistics and natural language processing based on naturally annotated big data: CCL NLP-NABD 2015* (Lecture Notes in Computer Science, Vol. 9427, pp. 389–400). Springer. https://doi.org/10.1007/978-3-319-25816-4_31
- [33] Zhu, T. (2022). Academic literature recommendation method integrating network representation learning and textual information. *Science & Technology Information*, 8(3), 81-92. <https://doi.org/10.3772/j.issn.2095-915X.2022>