



Content-Based Hybrid Deep Neural Network Citation Recommendation Method

Leipeng Wang^{1,2,3}, Yuan Rao^{1,2,3(✉)}, Qinyu Bian^{1,2,3},
and Shuo Wang^{1,2,3}

¹ Shenzhen Research Institute of Xi'an Jiaotong University,
Shenzhen 518057, China

raoyuan@mail.xjtu.edu.cn

² Lab of Social Intelligence and Complex Data Processing, Software School,
Xi'an Jiaotong University, Xi'an 710049, China

³ Shanxi Joint Key Laboratory for Artifact Intelligence (Sub-Lab of Xi'an
Jiaotong University), Xi'an 710049, China

Abstract. The rapid growth of scientific papers makes it difficult to query related papers efficiently, accurately and with high coverage. Traditional citation recommendation algorithms rely heavily on the metadata of query documents, which leads to the low quality of recommendation results. In this paper, DeepCite, a content-based hybrid neural network citation recommendation method is proposed. First, the BERT model was used to extract the high-level semantic representation vectors in the text, then the multi-scale CNN model and BiLSTM model were used to obtain the local information and the sequence information of the context in the sentence, and the text vectors were matched in depth to generate candidate sets. Further, the depth neural network was used to rerank the candidate sets by combining the score of candidate sets and multi-source features. In the reranking stage, a variety of Metapath features were extracted from the citation network, and added to the deep neural network to learn, and the ranking of recommendation results were optimized. Compared with PWFC, ClusCite, BM25, RW, NNRank models, the results of the Deepcite algorithm presented in the ANN datasets show that the precision (P@20), recall rate (R@20), MRR and MAP indexes rise by 2.3%, 3.9%, 2.4% and 2.1% respectively. Experimental results on DBLP datasets show that the improvement is 2.4%, 4.3%, 1.8% and 1.2% respectively. Therefore, the algorithm proposed in this paper effectively improves the quality of citation recommendation.

Keywords: Citation recommendation · Recurrent neural network · Convolutional neural network · BERT · Deep semantic matching

1 Introduction

With the rapid increase in the number of scientific literature, researchers have found that it is a very time-consuming task to track the latest developments in their field of research in time and find suitable references in the process of research and writing articles. The accumulation of literature reading is also affected by the inefficient and backward technology of existing search papers. Traditionally, the search for related

research papers based on keywords and search methods based on user interests and behavioral preferences has significant drawbacks in query efficiency and recommendation coverage. In the face of a large amount of academic literature, it is an urgent problem for workers to provide an efficient and high-quality personalized citation mining and recommendation that meets their current needs.

In order to solve the above problems, two kinds of citation recommendation strategies have been proposed: global recommendation and local recommendation. Global recommendation analyzes the characteristics of titles, abstracts, and authors in the target literature to mine relevant citations, so as to obtain the relevant literature. Local recommendation is a more fine-grained recommendation. It is more complicated to make citation recommendations based on a paragraph and context of the article. This article focuses on the research of global recommendation methods.

Recently, more and more people are focusing on deep learning-based methods to study citation recommendations. Bhagavatula et al. [22] put forward the NNrank model, which uses a supervised neural network to extract the semantic information of sentences in the paper, and recommend references with metadata features. But the semantic information extracted by such a simple neural network is not very rich, and the use of metadata makes the citation biased to self-citation.

In view of the above shortcomings of the existing methods, we use the Bert model to extract the high-level semantic representation vector in the text, and use the multi-scale CNN model and the Bilstm model to obtain the local information and the sequence information of the context in the sentence. Then, we match the generated text vector with deep semantics to improve the problem of insufficient text semantics in the candidate generation stage of NNrank model. Aiming at the problem that the NNrank model uses metadata to bias itself in the reordering phase, we propose a variety of different meta-path features from the citation network, add their similarities to the deep neural network to learn, optimize the ranking of the recommendation results, and finally return An important related citation recommendation list.

The main contributions of this article are as follows:

- 1) A new model of hybrid neural network model based on multiple strategies is proposed to extract deep semantic information and citation recommendations;
- 2) Use a deep neural network model to fuse multiple citation features to optimize the learning set for candidate sets to generate a high-quality academic paper citation recommendation result;
- 3) Perform in-depth experimental research on AAN and DBLP datasets, verifying the effectiveness of the proposed algorithm model.

The remainder of this paper is organized as follows. The second section introduces related work. The third section proposes feature extraction and task definition. The fourth section introduces specific model methods. The fifth section introduces experiments and results. The sixth section presents the conclusions of this paper.

2 Related Work

2.1 Global Citation Recommendations

Global recommendation analyzes relevant information such as titles, abstracts, and authors in the target literature to mine relevant citations, thereby obtaining comprehensive relevant literature. Global citation methods are mainly divided into three categories: collaborative filtering (CF), content-based filtering (CBF), and graph model-based citation recommendation methods.

Collaborative filtering (CF) based methods recommend citations based on ratings provided by other scientific researchers with similar research. Livne et al. [1] use collaborative filtering (CF) technology to research recommendation papers. Their work is based on citation network, which is a social network based on Citation relationship. They propose four types of collaborative filtering methods to recommend research papers, including Co-citation matching, User-Item CF, Item-Item CF and Bayesian Classifier. Yang et al. [2] proposed a collaborative filtering method based on sorting, assuming that users have similar reading interests in sorting common academic papers. Sugiyama et al. [3] extracted user research preferences from the list of published papers and cited papers, and constructed user research preferences. Users' preferences can be enhanced not only by papers published by researchers in the past, but also by papers cited by users. The content-based filtering (CBF) method uses vocabulary or topic features to determine when the paper is relevant to the needs of the researcher. Li et al. [4] proposed a conference paper recommendation method based on CBF. THE method extracts various pairwise features and applied pairwise learning to a rank model to predict papers that meet the preferences of the users.

The graph-based method considers it as a link prediction problem and solves it by using random walks. Gori et al. [6] constructed a homogenous citation graph and applied the PageRank algorithm to recommend scientific papers. Meng et al. [7] considered topics as particular nodes and build a four-layer heterogeneous publication graph, and then, they applied a random walk algorithm to recommend papers. Jardine et al. [8] extended the bias and transition probabilities of PageRank by considering topic distributions that were extracted from papers to predict scientific papers. Compared with ranking papers by whole link information on graphs, the node similarities on the sub-structures of a document network are much easier to compute, and they can reveal more explicit citation patterns. Sun et al. [9] introduced the concept of meta-path, which is a sequence of nodes in a network. They showed that meta-path-based score can obtain achievable performance for similarity search. Ren et al. [10] extracted various meta-path based features from citation graphs and proposed a hybrid model, called ClusCite, which combines nonnegative matrix factorization (NMF) with authority propagation. Guo et al. [11] extracted fine-grained co-authorship from citation graphs and recommended papers by graph-based paper ranking in a multi-layered graph. They further expanded the ranking approach with mutually reinforced learning for personalized citation recommendation. Mu et al. [12] expanded the ranking approach with mutually reinforced learning for personalized citation recommendation.

2.2 Distributed Representation of Texts

Distributed text representation refers to the method of using deep neural network algorithms to train vector representations of natural language objects (words, phrases, sentences, paragraphs, documents, etc.). Such vectors are also called text embedding vectors. Distributed representation vector is a low-dimensional dense vector learned from a large unsupervised corpus. Huang et al. [13] show that the distributed representation vector carries the semantic information of the text, which can be used as an effective expression of the text, applied to various natural language processing tasks, and achieved very excellent performance.

Bengio et al. [14] first applied distributed representation to statistical language models. This model is also known as neural network language model. In text representation learning, Word2Vec [15], GloVe [16], ELMo [17] and other models use small-scale data sets to train the semantic representation of text. Relative representation methods such as bag of words have achieved certain improvements in semantic representation. Devlin et al. [18] proposed a pre training model, Bert, which can effectively obtain the rich semantic information in the text and solve the problem of multi translation of one word in the text.

In the deep semantic matching task under the domain-specific data set, establishing the relationship model between texts is the most direct way to learn distributed representation. Huang et al. [19] proposed DSSM method, which uses DNN model to represent text as low latitude semantic vector in the representation layer to predict the similarity between sentences in the matching layer, and the model has achieved good results. However, DSSM uses the bag of words model (BOW), so it loses the word order information and context information. Hu et al. [20] proposed CLSM model using CNN method, extracting context information under sliding window through convolution layer, and extracting global context information through pooling layer, so that context information can be effectively retained. However, it is difficult to retain the context information effectively. To solve the problem that CLSM model cannot capture remote context features, Palangi et al. [21] uses BiLSTM method to perform semantic matching task. The above method has achieved good results in the paired task, but in the citation recommendation task, an important problem of pairwise matching is to enumerate and calculate the relationship between each pair. Bhagavatula et al. [22] proposed the NNrank model, which uses a supervised neural network to extract the semantic information of the sentences in the paper, and recommends the references based on the characteristics of the metadata. But the semantic information extracted by such a simple neural network is not very rich.

In view of the above problems, this paper proposes a method using hybrid deep neural networks for citation recommendation tasks. Not only can the problem of insufficient rich semantic information extraction in the existing model be solved, but also various meta-paths are extracted from the citation network, which solves the problems of single citation recommendation features and optimizes the ranking of recommendation results.

3 Feature Extraction and Task Definition

3.1 Feature Extraction

This paper first uses supervised neural networks to extract valid citation features from scientific literature data. Ren et al. [10] proposed many citation features, and checked their validity in the experiments in this paper, and selected useful citation features to add to the recommended scheme in this paper. The references of academic papers contain rich meta-path information. This paper extracts meta-path features from citation heterogeneous networks and adds them to recommendation schemes to further enrich the feature information of citations in this article. In this paper, we extract 22 kinds of multi-source features from citation data, which are mainly divided into three categories: text features, metadata features, meta-path features.

Text Features: Generate embedded vectors from titles, abstracts, keywords, and meeting names in documents using the BERT model.

Metadata Features: Paper cited count, which measures the impact of the paper, users always tend to cite more articles. The number of times the author has been quoted, which measures the influence of the author. In this paper, the author’s average number of citations is calculated. Author similarity Jaccard (authors), the feature of which is to calculate the Jaccard similarity of author index in citation pairs, is mainly used to measure the inclusion of similar authors in recommended papers, because these collaborators usually work in the same research field.

Meta-path Similarity: We extract various meta-path based features from the dataset. We select 15 different meta-paths, including $PAAP$, $PAVP$, $PVAP$, $(PXP)^y$, $PXP \rightarrow P$, $PXP \leftarrow P$ where $X = \{A, V, T\}$, and $y = \{1, 2\}$. We choose both PathSim [23] and a random-walk based measure [24] to calculate the meta-path-based features, since PathSim can only be applied for symmetric meta-paths.

3.2 Task Definition

The citation recommendation model DeepCite proposed in this paper is defined as a two-stage task, as shown in Fig. 1.

Task 1: Candidates Generation

This paper uses a supervised hybrid neural network to train text features (title, abstract) in academic papers to generate distributed representation vectors for deep semantic matching, and then sorts the candidate sets it generates to filter and quickly generate citations for query documents d_q Set $O(1000)$.

Task 2: Reranking

According to the candidate score $s_1(d_q, d_i)$ between the query and candidate documents in the citation candidate set O , combined with the 22 citation features (text, metadata, meta path similarity) extracted from the citation dataset, re-estimate the citation document (d_q, d_i) to sort the citation probabilities, and finally return the candidate citation with the highest citation probability for the query document d_q .

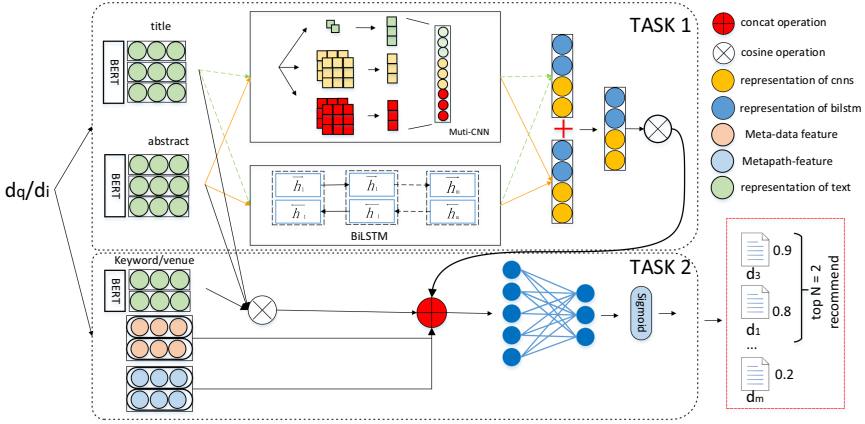


Fig. 1. The structure design of deepCite model. In task 1, we extract the text features from the title and summary to generate candidate sets through semantic matching. In task 2, we use multi-source features to reorder the candidate set and return the candidate papers with high scores for users

4 Content-Based Hybrid Deep Neural Network Citation Recommendation

In this section, we will introduce in detail how to use deep hybrid neural network to return a list of reference list top-N for users based on a given query document d_q . As shown in Fig. 1, this paper first uses the BERT multi-layer feature representation method to context embed words, then uses CNNs with different convolution kernels to extract local features of different scales, and then uses BiLSTM to strengthen the sequence relationship between words. Finally, the generated document representation vector is subjected to deep semantic matching to generate a candidate set. Further, a four-layer feedforward neural network is used to combine the candidate set score and multi-source features to learn, and finally the sigmoid function is used to output the recommended citation probability.

4.1 Candidates Generation

BERT is one of the pre-trained models with the highest performance in the learning field of NLP. The text uses the data in the scientific literature to fine-tune the original BERT model, so that the model parameters can be better adapted to the scientific literature citation recommendation field.

First, a pair of queries and candidate documents (d_q, d_i) title $d[\textit{title}]$ and abstract $d[\textit{abstract}]$ are input into the fine-tuned BERT model to generate a high-level semantic representation of the text vector $v(w_1, w_2, \dots, w_n)$.

Where n represents the number of words, the self-attention mechanism used by the BERT model solves the problem of long-distance dependence, but indicates that the

semantic information is a shallow representation, and then uses the size CNN and BiLSTM to extract richer semantic information.

Recently, CNN-based research has achieved excellent performance in the fields of text classification, named entity recognition, etc., because it can extract word-rich n-gram information to extract local features. In this paper, a grouped convolutional network is designed to extract local features at different scales. The first group is a convolutional neural network with a convolution kernel size of 1, the second group is a convolutional neural network with a convolution kernel size of 3, and the third group is also a convolutional neural network with a convolution kernel size of 3. The difference between the two groups is that they have different numbers of channels.

The sentence s_k is used as a matrix $s_k \in \mathbb{N}^{t \times m}$, where t represents the dimension of the word vector and m is the number of words in the sentence s_k . The word vector of the i -th word of the sentence s_k is expressed as $v(w_i) \in \mathbb{N}^t$. $v(w_i : w_{i+j})$ is used to represent the word vector $[v(w_i), \dots, v(w_j)]$, the convolution kernel of the convolution operation is $W^h \in \mathbb{N}^{l \times ht}$, where h is the size of the convolution window, and the feature vector $c_i \in \mathbb{N}^l$ is generated. The definition formula is as follows:

$$e_i = f(w^h \times v(w_i : w_j) + b) \quad (1)$$

Among them, $b \in \mathbb{N}^{ht}$ has a bias term, and f is a non-linear function. In this paper, the tanh function is used. The convolution window contains all the word structures $w_1 : w_h, w_2 : w_{h+1}, \dots, w_{m-h} : w_m$ and the resulting vector is:

$$e_i = [e_1, e_2, \dots, e_{m-h+1}] \quad (2)$$

Each convolutional neural network is activated using the ReLU function. Finally, the local features of different scales extracted by the packet convolution network are stitched together to obtain a high-level semantic representation:

$$e_{d(CNN)} = e_{d(1-CNN)} : e_{d(2-CNN)} : e_{d(3-CNN)} \quad (3)$$

CNN performs well in many tasks, but the biggest problem is the fixed field of view of filter size. On the one hand, it is impossible to model longer sequence information. On the other hand, the adjustment of filter sizes hyperparameters will be cumbersome. In natural language processing, the BiLSTM model can better express the following information. Adding the BiLSTM module to the model can strengthen the sequence relationship between words and make the words have semantic information in upper and lower languages. BiLSTM controls contextual information through forget gate, input gate, and output gate.

$$\begin{aligned} f_t &= \sigma(w_f[h_{t-1}, x_f] + b_f) \\ i_t &= \sigma(w_i[h_{t-1}, x_f] + b_i) \\ \tilde{C}_t &= \tanh(w_c[h_{t-1}, x_f] + b_c) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C} \end{aligned} \quad (4)$$

Where σ is the sigmoid function, w_f is the forgetting gate weight, b_f is the forgetting gate bias, i_t is the updated weight, and the tanh layer generates a new input vector. Then the output weights and output values are O_t and h_t , respectively, and the calculation formula is:

$$\begin{aligned} O_t &= \sigma(w_0[h_{t-1}, x_t] + b_o) \\ h_t &= O_t * \tanh(C_t) \end{aligned} \quad (5)$$

The forward and backward outputs are stitched together as the output vector of the BiLSTM model:

$$e_{d(BiLSTM)} = h_i^{before} : h_i^{back} \quad (6)$$

Finally, the CNN convolution and BiLSTM output vectors are stitched together as the final output vector of the presentation layer:

$$e_{d[field]} = e_{d[CNN]} : e_{d[BiLSTM]} \quad (7)$$

In this article, title and abstract are used to represent the document vector of the paper, where μ^{title} title is a scalar model parameter:

$$e_d = \alpha^{title} \frac{e_{d[title]}}{\|e_{d[title]}\|_2} + \alpha^{abstract} \frac{e_{d[abstract]}}{\|e_{d[abstract]}\|_2} \quad (8)$$

This article uses cosine similarity to calculate the similarity between query and candidate document doc, that is, the candidate score.

$$s_1(d_q, d_i) = \cos ine(e_{d_q}, e_{d_i}) \quad (9)$$

In this paper, the triples $\tau\langle d_q, d^+, d^- \rangle$ are used to establish model learning parameters, where d_q is the query document, d^+ is the referenced document, and d^- is the unreferenced document. In training, the model is trained using the Triplet Loss function to predict high cosine similarity for the (d_q, d^+) combination and low cosine similarity for the (d_q, d^-) combination.

$$loss = \max(\gamma\alpha(d^-) + s_1(d_q, d^+) + B(d^+) - s_1(d_q, d^-) - B(d^-), 0) \quad (10)$$

In training, using the method proposed by Bhagavatula C et al. [22], Using random negative sampling, $\alpha(d^-) = 0.3$, randomly sampling papers that are not cited by d_q . The lifting function is defined as:

$$B(d) = \frac{\sigma(\frac{d[in-citation]}{100})}{50} \quad (11)$$

Where σ is the sigmoid function $d[in - citation]$ is the number of times the papers have been cited in the citation dataset.

4.2 Reranking

In task 2, the text features, metadata features, and meta-path similarity features of the candidate document doc are extracted from the citation network according to the candidate set generated in task 1. Then a four-layer feedforward neural network is used to reorder the candidate set. Finally return a list of TopN citation recommendations for users.

As shown in Fig. 1, input the title, abstract, keyword, and course into the BERT to get the embedding vector, and use the cosine similarity to calculate the text feature similarity:

$$g_{field} = cosine(e_{dq[field]}, e_{di[field]}) \quad (12)$$

Then the text similarity, meta-path feature similarity features, etc. are spliced into a feedforward neural network, which is defined as follows:

$$\begin{aligned} s_2(dq, di) &= FeedForward(h) \\ h &= [g_{title}; g_{abstract}; g_{venue}; g_{keyword}; \\ &\quad Jacard(authors); doc_{citations}; aut_{citations}; \\ &\quad Metapath(PxP); s_1(dq, di)] \end{aligned} \quad (13)$$

Where FeedForward is a four-layer feedforward neural network. The first three layers are the linear unit ReLU function and the last layer is the Sigmoid function. $Metapath(PxP)$ is the concatenation of the similarities of 15 different metapaths. $doc_{citations}$ refers to the number of times a candidate document should be cited. $aut_{citations}$ is the average number of times a paper publish by the candidate's author is cited. The index of $Jacard(authors)$ similarity between co-authors of the paper, $s_1(dq, di)$ is the candidate set score in task 1.

5 Experiments and Results

5.1 Evaluation Metrics

To evaluate the quality of the recommendations, we use the citation information of the training papers to train our model, and the reference lists of the testing papers are used as the ground truth. Following common practice, we employ the following evaluation metrics.

- Recall

The recall is defined as the ratio of the truly cited literature in the recommendation list to the reference of the test literature. The recall rate is an important indicator

related to the evaluation of recommendations. The formula for calculating the recall rate is as follows:

$$Recall = \frac{\sum_{d \in Q(D)} |R(d) \cap T(d)|}{\sum_{d \in Q(D)} |T(d)|}$$

- Precision

Accuracy is the ratio of the number of documents retrieved to the total number of documents retrieved. It measures the accuracy of the retrieval system. Accuracy is an important indicator of the performance of the recommendation system. The calculation formula for accuracy is as follows:

$$precision = \frac{\sum_{d \in Q(D)} |R(d) \cap T(d)|}{\sum_{d \in Q(D)} |R(d)|}$$

- Mean Reciprocal Rank (MRR)

MRR refers to the ranking of the standard answer in the results given by the evaluated system, taking the reciprocal as its accuracy, and averaging all the questions. The MRR calculation formula is as follows:

$$MRR = \frac{1}{|Q(D)|} \sum_{d \in Q(D)} \frac{1}{rank_d}$$

Where $rank_d$ is the position of the first correct result in test set D.

- Mean Average Precision (MAP)

MAP is the average of a set of average accuracy rates (AP). The calculation method of AP is as follows:

$$AP_j = \sum \frac{p(i) \times pos(i)}{\text{number of positive instance}}$$

Where i is the ranking position in the result queue, and $P(i)$ is the accuracy of the first i results.

$$P(i) = \frac{\text{Related documents}}{\text{Total number of documents}}$$

Both MRR and MAP account for the rank of the recommended citation, and consequentially, it heavily penalizes the retrieval results when the relevant citations are returned at low rank.

5.2 Datasets

To evaluate our proposed model, we choose two bibliographic datasets, AAN and DBLP, which have different sizes of research publications in different research fields.

ANN dataset¹: Radev et al. established the ACL Anthology Network (AAN) dataset, which contains full text information of conference and journal papers in the computational linguistics and natural language processing field. We use a subset of a 2012 release that contains 13,885 papers published from 1965 to 2012. For evaluation purposes, we divide the entire dataset into two disjoint sets, where papers published before 2012 are regarded as the training set (12,762 papers) and the remaining papers are placed in the testing set (1,123 papers).

DBLP dataset²: DBLP is a well-known online digital library that contains a collection of bibliographic entries for articles and books in the field of computer science and related disciplines. We use a citation dataset that was extracted and released by Tang et al. [27]. This article does not use a complete data set, but chooses a subset because some examples lack complete references. We divide papers published before 2009 into training sets (29193 papers) and papers published from 2009–2011 into Test set (2869 papers) (Table 1).

Table 1. Statistics of ANN and DBLP

		Papers	Authors	Venues	Citations
ANN	Train	12762	9766	467	68475
	Test	1123	1557	32	10437
DBLP	Train	29193	32541	751	397316
	Test	22869	3184	104	31501

5.3 Comparison with Other Approaches

In this experiment, a pre-trained BERT (12 layers, 768 hidden, 12 heads, 110M parameters) is used, and its fine-tuning is applied to the ANN and DBLP datasets. In the experiment, the length of the title of each paper is set to 50, the length of the abstract is set to 512, the maximum length of the keyword is 20, and the maximum number of citations is 100. The hidden layer of BiLSTM is 768, and the size of CNN is set to $1 \times 1, 3 \times 3, 3 \times 3$. During the training process, Adam’s method was used to optimize the parameters of the model. The learning rate and dropout were set to $1e-5$ and 0.55 in task 1, and set to $1e-4$ and 0.5 in task 2. All experiments in this article are Linux CentOS The 7.6.1810 system is completed on the open source framework Pytorch1.0, which has an NVIDIA TITAN X graphics card (12G).

In order to verify the improvement of the performance of the evaluation index proposed by the algorithm proposed in this paper, the ANN and DBLP data sets are collected on Recall (R@20, R@40), Precision (P@10, P@20), MAP and MAP indicators. Algorithms for comparison:

- ClusCite [10]: ClusCite assumes that citation features should be organized into different groups, and each group contains its own behavior pattern to represent the

¹ <http://clair.eecs.umich.edu/ann/>.

² <http://dblp.uni-trier.de/db/>.

research interest. This method combines NMF and network regularization, to learn group and authority information for citation recommendation. To ensure fairness in the comparison, we use all extracted citation features in this paper for ClusCite.

- TopicSim: We use the original PLSA to derive the topic information; then, we recommend papers that have high topic relevance with the query.
- BM25 [25]: BM25 is a well-known ranking method for measuring the relevance of matching documents to a query based on the text. We calculate the text similarity between the papers by using both TF and IDF for BM25.
- PWFC [26]: PWFC uses a fine-grained cooperative relationship between authors to build a three-layer graph after ranking based on a random walk method.
- NNselect [22]: NNselect algorithm proposes a supervised neural network that trains the title and abstract of the paper into a low-latitude dense representation vector of text, and further uses ANN neighbors to make recommendations.
- NNrank [22]: It returns the candidate set results based on the NNselect method, and then uses a three-layer feedforward neural network to sort. In this article, we choose to use Metadata data for experiments.

The experimental results are shown in Table 2. It is possible to observe the tendency of these methods to evaluate the accuracy of the indicators. The recommendation results of the PWFC and TopSim algorithms are relatively poor, because there are certain limitations in recommending citations only through citation relationships, content relevance, and relationships between co-authors. In the experiments, the NNselect algorithm is superior to models such as TopicSim in all indicators, which indicates that text-based distributed representation is richer than text-based semantic features extracted from topic-based model features. Based on the NNselect model, NNrank uses Metadata features and deep neural networks to reorder the results of the candidate set. Experimental data shows that all index data of NNrank is better than the NNselect model, proving the important influence and effect of Metadata features on the recommendation results.

In this paper, a pre-trained model BERT is used, and its fine-tuning is applied to the ANN and DBLP scientific literature data sets. The BERT model has achieved outstanding results in multiple NLP tasks, and the text vectors it outputs are obtained using multi-scale CNN and BiLSTM models to obtain local And sequence information, and then perform deep semantic matching. Compared with the NNselect model, the DeepCite model proposed in this paper is a hybrid neural network, and its proposed semantic information is more abundant and effective. This paper uses 22 features to reorder the candidate results in task 2, and solves the problems of citation recommendation features. Experimental results prove that the DeepCite method is better than the current best model. Compared with the NNrank model and the DeepCite model in the ANN dataset, In the indicators P @ 10, P @ 20, R @ 20, R @ 40, MAP, and MRR, they increased by 2%, 2.3%, 3.9%, 3.4%, 2.4%, and 2.1%. Compared with the NNrank model in the DBLP dataset, it has increased by 2.7%, 2.4%, 4.3%, 3.2%, 1.8%, and 1.4%, respectively.

Table 2. Performance comparison between different methods on ANN and DBLP datasets

Dataset	Method	P@10	P@20	R@20	R@40	MAP	MRR
ANN	BM25	0.168	0.105	0.161	0.287	0.117	0.335
	TopSim	0.118	0.066	0.188	0.188	0.089	0.154
	PWFC	0.205	0.155	0.228	0.196	0.170	0.458
	ClusCite	0.251	0.193	0.272	0.347	0.188	0.562
	NNselect	0.297	0.241	0.376	0.481	0.211	0.582
	NNRank	0.354	0.259	0.403	0.522	0.233	0.690
	DeepCite	0.374	0.282	0.442	0.556	0.257	0.711
DBLP	BM25	0.174	0.125	0.182	0.308	0.127	0.349
	TopSim	0.091	0.079	0.128	0.218	0.157	0.095
	PWFC	0.214	0.169	0.233	0.332	0.177	0.461
	ClusCite	0.261	0.195	0.302	0.383	0.198	0.548
	NNselect	0.287	0.230	0.363	0.472	0.205	0.576
	NNRank	0.345	0.247	0.390	0.517	0.227	0.689
	DeepCite	0.372	0.271	0.433	0.549	0.245	0.703

5.4 Ablation Experiment Results and Analysis

In this paper, ablation experiments are used to prove the feasibility and effectiveness of each module in the DeepCite method of global citation recommendation model. In the experiment, the minimum number of citations for the papers on the ANN and DBLP datasets was set to five, and the experiments were compared with other parameters fixed.

As shown in Table 3, BERT means that in the ablation experiment, only the BERT model is used to perform semantic matching in the candidate set generation task, and then a citation is recommended for the user according to the result of the matching. The BERT_MCNN model refers to the task of generating candidate sets, inputting text vectors generated by BERT into multi-size CNNs, and then performing semantic matching, and then recommending citations for users based on the results of the matching. The BERT_BiLSTM model refers to the task of generating candidate sets, inputting text vectors generated by BERT into a bidirectional long-term and short-term memory network BiLSTM, and then performing semantic matching, and then recommending citations for users based on the results of the matching. The BERT_MCBL model refers to the task of generating candidate sets. The text vectors generated by BERT are simultaneously input into multi-size CNN and BiLSTM, and the sentence vectors of their data are further stitched for semantic matching, and then citations are recommended for users based on the matching results. In this paper, ablation experiments are used to prove the feasibility and effectiveness of each module in the candidate set generation DeepCite method, and compared with all parameters fixed.

As shown in Figs. 2, with the increase of modules, the candidate set generation module has been improved on these indicators P@10, P@20, R@20, R@40, which verifies that the model is a candidate for each module. The solution generation process has been improved, which verifies that this model has played an effective role in the

Table 3. DeepCite model ablation experiment on Ann and DBLP datasets

Dataset	Method	P@10	P@20	R@20	R@40	MAP	MRR
ANN	BERT	0.294	0.244	0.378	0.490	0.490	0.584
	BERT_MCNN	0.308	0.248	0.385	0.496	0.496	0.595
	EERT_BiLSTM	0.302	0.246	0.383	0.494	0.494	0.592
	BERT_MCBL	0.322	0.258	0.391	0.519	0.519	0.601
	DeepCite	0.374	0.282	0.442	0.556	0.556	0.711
DBLP	BERT	0.291	0.239	0.371	0.482	0.206	0.582
	BERT_MCNN	0.306	0.251	0.390	0.503	0.223	0.595
	EERT_BiLSTM	0.298	0.246	0.381	0.494	0.216	0.591
	BERT_MCBL	0.315	0.252	0.399	0.513	0.232	0.598
	DeepCite	0.372	0.271	0.433	0.549	0.245	0.703

process of candidate solution generation during the various modules. Based on the BERT model, this paper uses three sets of CNNs of different sizes to obtain the local semantic information of the text, which improves the effect of deep semantic matching. Then the BiLSTM model is used to obtain the sequence information in the text, which further improves the effect of semantic matching and generates high Quality citation candidate set.

The experimental results are shown in Table 3. Compared with the BERT model, the BERT-BL3C model proposed in task 1 in the candidate set generation stage has an accuracy rate (P@10, P@20) and a recall rate (R@20, R@40), the average reciprocal ranking (MRR), and the average accuracy rate (MAP) increased by 2.4%, 1.3%, 2.8%, 3.1%, 2.6%, and 1.6%, respectively. Increased 2.8%, 1.4%, 1.1%, 2.9%, 2.2%, 1.8% on the ANN dataset.

In Task 2, the candidate set was reordered by combining 22 citation features. Compared to BERT-BL3C, the DeepCite model has an accuracy rate (P @ 10, P @ 20) and a recall rate (R @ 20, R @ 40) on the ANN dataset. 40), the average reciprocal ranking (MRR), the average accuracy rate (MAP) increased by 5.2%, 2.4%, 5.1%, 3.7%, 2.6%, 11%, respectively, on the DBLP dataset increased by 5.7%, 1.9%, 3.4%, 3.6%, 1.3%, 10.5%.

Meta-path Validity

In order to solve the problem that the single feature in the traditional citation recommendation model leads to the poor effect of recommendation ranking, our adds 15 kinds of meta-path similar features to the deep neural network model to improve the ranking effect of the model. In order to prove its effectiveness, two different DeepCite models were trained on different numbers of recommended citations, one using meta-path similarity features and the other not using meta-path similarity features. The experimental results on the DBLP dataset prove that this feature effectively improves the mean to number ranking (MRR) during the reordering stage, as shown in Fig. 3.

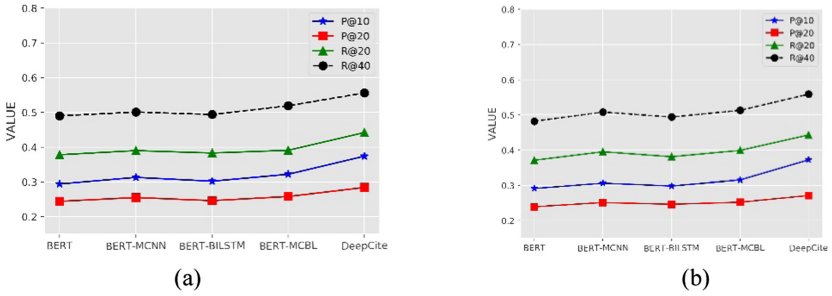


Fig. 2. Ablation experiment on DBLP and ANN datasets.

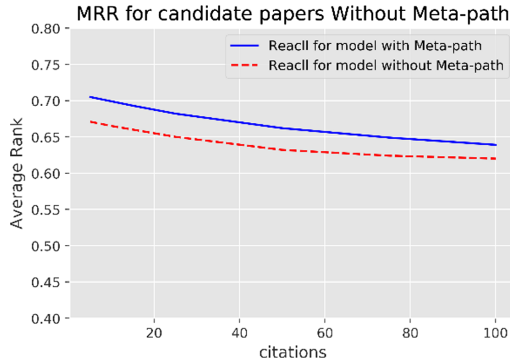


Fig. 3. MRR of predictions with varying number of candidates

Effect of Paper Citation Frequency

The minimum number of paper citations in the paper dataset used in model comparison experiments and model ablation experiments is 5. As shown in Fig. 4, we study the impact of the paper citation frequency on the performance of citation recommendation. In the ANN and DBLP datasets, the citation frequency is set between 1 and 7, respectively, at the accuracy rate (Precision@10, Precision@20). Experiments with recall rates (Recall @ 20, Recall @ 40). From the analysis of experimental results, as the citation frequency increases, the performance of our proposed model improves. When the citation frequency is 5, the global citation recommendation Model has the best recommendation performance. In other experiments in this paper, the reference frequency is set to 5. Generally speaking, uncited papers are not used for learning, so they can be treated as sparse data during testing. Furthermore, the experiment data can be improved according to the frequency of citations to establish a fully functional high-performance global citation recommendation model and provide users with recommendation services.

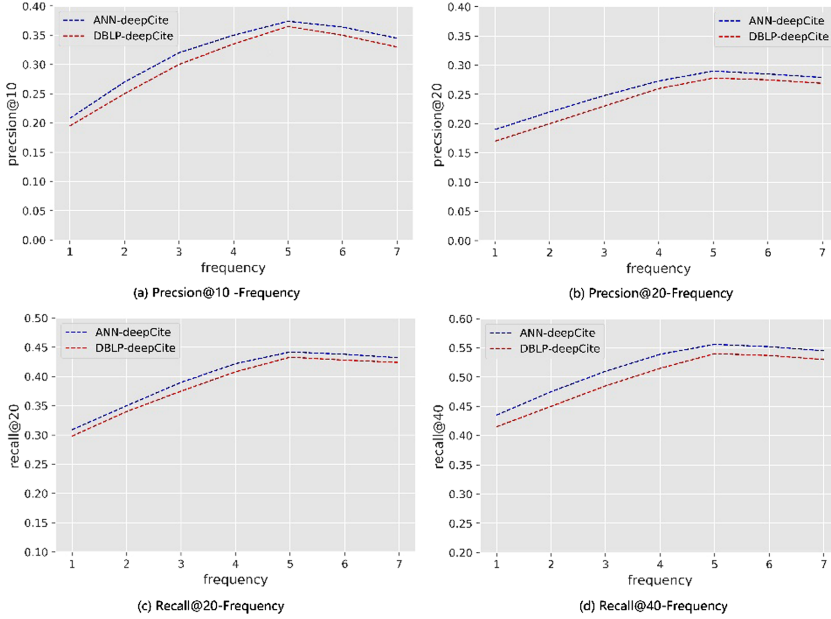


Fig. 4. Impact of paper citation frequency

6 Conclusions

This paper studies citation recommendations from two aspects: citation content and deep learning. A hybrid deep neural network model is used to extract high-level semantic representation vectors of text, and deep semantic matching is performed on query documents and candidate documents to generate candidate sets.

At the same time, the document representation vector generated by task 1 can be embedded in the vector space to improve the text matching speed. This paper extracts various effective citation features in the citation network, and uses the deep neural network to reorder the results of the candidate set in combination with the candidate set score, thereby improving the performance of citation recommendation. The experimental results show that the proposed DeepCite algorithm effectively changes the current citations. The algorithm is inefficient and the recommendation quality is low. The limit is that meta-path information of the citation network is not studied in this article. The next work will focus on the construction of complex citation networks. The deep learning technology will be used to embed the nodes and path information in the citation network.

Acknowledgment. The research work is supported by “Shenzhen Science and Technology Project” (JCYJ20180306170836595); “National key research and development program in China” (2019YFB2102300); “the World-Class Universities (Disciplines) and the Characteristic Development Guidance Funds for the Central Universities of China” (PY3A022); “Ministry of Education Fund Projects” (No. 18JZD022 and 2017B00030); “Basic Scientific Research

Operating Expenses of Central Universities” (No. ZDYF2017006); “Xi’an Navinfo Corp.& Engineering Center of Xi’an Intelligence Spatial-temporal Data Analysis Project” (C2020103); “Beilin District of Xi’an Science & Technology Project” (GX1803).

References

1. Livne, A., Gokuladas, V., Teevan, J., et al.: CiteSight: supporting contextual citation recommendation using differential search. In: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 807–816 (2014)
2. Yang, C., et al.: CARES: a ranking-oriented CADAL recommender system. In: Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries, pp. 203–212. ACM (2009)
3. Sugiyama, K., Kan, M.: Scholarly paper recommendation via user’s recent research interests. In: Proceedings of the 10th Annual Joint Conference on Digital Libraries, pp. 29–38. ACM (2010)
4. Li, S., et al.: Conference paper recommendation for academic conferences. *IEEE Access* **6**, 17153–17164 (2018)
5. Torres, R., McNee, S.M., Abel, M., et al.: Enhancing digital libraries with TechLens. In: Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries, pp. 228–236. ACM (2004)
6. Gori, M., Pucci, A.: Research paper recommender systems: a random-walk based approach, In: *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 778–781. IEEE (2006)
7. Meng, F., et al.: A unified graph model for personalized query oriented reference paper recommendation. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, pp. 1509–1512. ACM (2013)
8. Jardine, J., Teufel, S.: Topical PageRank: a model of scientific expertise for bibliographic search. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 501–510 (2014)
9. Sun, Y., et al.: Pathsim: meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* **4**(11), 992–1003 (2011)
10. Ren, X., et al.: ClusCite: effective citation recommendation by information network-based clustering, In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 821–830 (2014)
11. Guo, L., et al.: Exploiting fine-grained co-authorship for personalized citation recommendation. *IEEE Access* **5**, 12714–12725 (2017)
12. Mu, D., et al.: Query-focused personalized citation recommendation with mutually reinforced ranking. *IEEE Access* **6**, 3107–3119 (2018)
13. Huang, E.H., Socher, R., Manning, C.D., et al.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, pp. 873–882. Association for Computational Linguistics (2012)
14. Bengio, Y., Ducharme, R., Vincent, P., et al.: A neural probabilistic language model. *J. Mach. Learn. Res.* **3**(Feb), 1137–1155 (2003)
15. Mikolov, T., Sutskever, I., Chen, K., et al.: Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119 (2013)

16. Ebesu, T., Fang, Y.: Neural citation network for context-aware citation recommendation. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1093–1096. ACM (2017)
17. Peters, M.E., Neumann, M., Iyyer, M., et al: Deep contextualized word representations. arXiv preprint [arXiv:1802.05365](https://arxiv.org/abs/1802.05365) (2018)
18. Devlin, J., Chang, M.W., Lee, K., et al.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
19. Huang, P.S., He, X., Gao, J., et al.: Learning deep structured semantic models for web search using click through data. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, pp. 2333–2338. ACM (2013)
20. Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: Proceedings of the NIPS, pp. 2042–2050 (2014)
21. Palangi, H., et al.: Semantic modelling with long-short-term memory for information retrieval. arXiv preprint [arXiv:1412.6629](https://arxiv.org/abs/1412.6629) (2014)
22. Bhagavatula, C., Feldman, S., Power, R., et al.: Content-based citation recommendation. arXiv preprint [arXiv:1802.08301](https://arxiv.org/abs/1802.08301) (2018)
23. Sun, Y., Han, J., Yan, X., et al.: Pathsim: meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* **4**(11), 992–1003 (2011)
24. Lichtenwalter, R.N., Lussier, J.T., Chawla, N.V.: New perspectives and methods in link prediction. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 243–252. ACM (2010)
25. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: Croft, B.W., van Rijsbergen, C.J. (eds.) *SIGIR 1994*, pp. 232–241. Springer, London (1994). https://doi.org/10.1007/978-1-4471-2099-5_24
26. Guo, L., Cai, X., Hao, F., et al.: Exploiting fine-grained co-authorship for personalized citation recommendation. *IEEE Access* **5**, 12714–12725 (2017)
27. Tang, J., Zhang, J., Yao, L., et al.: Arnetminer: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 990–998. ACM (2008)