Path-Based Academic Paper Recommendation

Shengjun Hua[1], Wei Chen[1,2], Zhixu Li[1], Pengpeng Zhao[1], and Lei Zhao[1](✉)

[1] School of Computer Science and Technology, Soochow University, Suzhou, China
20184227008@stu.suda.edu.cn,
{robertchen,zhixuli,ppzhao,zhaol}@suda.edu.cn
[2] Institute of Artificial Intelligence, Soochow University, Suzhou, China

Abstract. The overabundance of academic papers makes it difficult for researchers to find relevant or interested papers. To address the problem, existing studies have developed many approaches to recommend academic papers effectively. Most of them mainly utilize content-based filtering or citation analysis to measure similarity or relatedness of two papers and recommend relevant papers to the given query. However, these recommended papers are usually discrete from each other, i.e., the relationship between recommended papers are omitted, which disables researchers from having an sight into the time-oriented development of the topic they are interested in. To overcome the drawbacks of existing work, we propose a novel academic paper recommendation method called PAPR (Path-based Academic Paper Recommendation). Our method aims to recommend an ordered path of relevant papers, which are of great benefit in helping researchers understand the development of a specific topic. During process, we take both content and network structure into account to learn the representation of a paper. Next, the similarity between papers are measured based on the representation. The experimental results based on real data show that the proposed method outperforms the state-of-art methods.

Keywords: Paper recommendation · Representation learning · Citation relations.

1 Introduction

Due to the overwhelming amount of academic papers published every year, researchers need to spend much time on searching relevant or interested papers, i.e., recommending academic papers becomes a challenging task. To address the problem, many digital libraries and recommender systems have been developed to recommend relevant literature according to the provided keywords or users' profiles. Although search engines accelerate search process, finding satisfactory scientific articles is still time-consuming and inconvenient.

A number of techniques have been proposed to recommend academic papers. Content-based filtering (CBF) [11] views content of document as keywords to

Z. Huang et al. (Eds.): WISE 2020, LNCS 12343, pp. 343–356, 2020.
https://doi.org/10.1007/978-3-030-62008-0_24

measure similarity between papers. Collaborative filtering (CF) is a classical recommendation method which recommends items to users based on others who have similar preferences [13]. Citation-based analysis uses citation relationships to calculate relatedness among papers [16]. Hybrid methods have been proposed to improve the recommendation results, which combine two or more recommendation techniques to get better performance [6].

The techniques mentioned above are mostly used to recommend academic papers that are similar to a given paper in some respects. As shown in Fig. 1(a), given a paper u_1 in the field of graph embedding, the recommended papers are u_6, u_2, u_4, and u_8. In these recommended papers: u_6 surveys the study of dimensionality reduction, which has the same background knowledge with u_1; papers u_2 and u_4 are topically related to u_1 and focus on addressing problem of network representation; paper u_8 provides a new fundamental theory which can be transferred to deal with the research problem in u_1. Existing recommendation methods rank papers based on relevance and return a set of discrete papers relevant to the given query. However, for researchers who concentrate on learning existing literature about the topic of the given paper, they want to know how the topic develops and evolves, and understand what it originates from. Conventional academic paper recommendation ignores the relationships of recommended papers and can not provide effective results to help researchers trace the development of the topic.

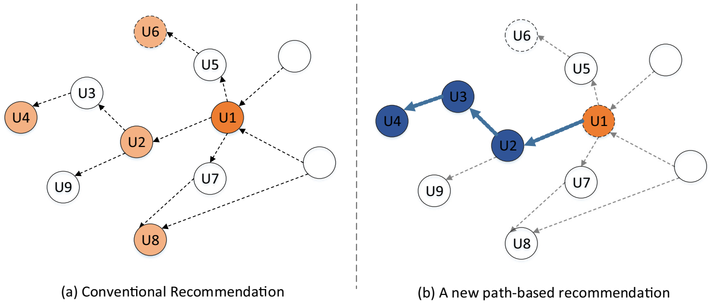

U6
U5
U3
U4
U1
U2
U9
U7
U8
(a) Conventional Recommendation
U6
U5
U3
U4
U1
U2
U9
U7
U8
(b) A new path-based recommendation

Fig. 1. Comparison of conventional recommendation and path-based recommendation.

To address above-mentioned problem, we propose a novel method called PAPR (Path-based Academic Paper Recommendation), which considers time-oriented development of the topic of the given paper, and recommends a sequence of academic papers. Figure 1(b) illustrates how our method works. Our method takes a paper u_1 as input and produces a 3-hop path u_2, u_3, u_4 from the paper. The proposed method concentrates on creating a deep and consecutive paper path. Nodes on the path are papers highly relevant and important, which represent the evolution and development of the paper's topic. Addressing this task will be beneficial to researchers who start to investigate a new topic.

In this paper, we take both content and network information into account to obtain better performance on paper recommendation. We learn the textual vector for each paper based on the semantic information (e.g., title and abstract) and utilize the network information to get relatedness of papers. Then, we combine semantic similarity and network relatedness to obtain the united similarity between papers. To get the path of recommendation, we use beam search [18] to get papers which have high relevance with all papers in the existing path.

The main contributions of this paper can be summarized as follows:

- We introduce a novel academic paper recommendation method to trace the origin and development of the topic of the given paper, which is beneficial for people to have an sight into the development of a research topic.
- We propose a novel strategy to measure the relevance between two papers and generate a consecutive paper path for recommendation, where both text semantic information and network structural information are considered.
- We conduct extensive experiments on two real-world datasets DBLP and ACM, and the results show that our proposed approach performs better than all compared methods.

The remaining sections of this paper are as follows. Section 2 presents the concept of scientific network and the problem definition. The overview of the path-based paper recommendation is introduced in Sect. 3. Section 4 describes the experimental setup and discusses the experimental results. Section 5 presents related work on recommending research papers. Section 6 concludes the paper.

2 Problem Definition

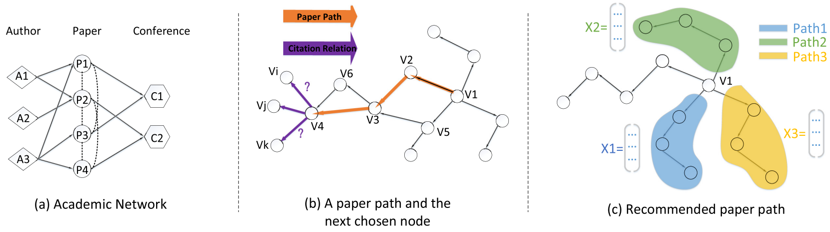
In this section, we introduce the concept of scientific network and then formally define the problem. The notations we will use are summarized in Table 1.

Table 1. Notations and explanations.

Notation	Explanation
G	Academic Network
V	Vertices in the graph G
E	Edges in the graph G
T_V	A set of object types
T_E	A set of relation types
Φ	Mapping function to embed vertices
N_i	Neighbors cited by node v_i

Definition 1. *(Academic Network). An academic network is defined as a graph $G = (V, E)$, where $V = \{v_1, ..., v_n\}$ represents n vertices and E is the set of edges in G. Each node v and each link e are associated with their mapping functions $\psi(v) : V \rightarrow T_V$ and $\varphi(e) : E \rightarrow T_E$, respectively. T_V and T_E denote the sets of predefined objects and relation types. A mapping function $\Phi : V \rightarrow \mathbb{R}^d\,(d \ll |V|)$ is used to learn the latent representation of papers.*

Example 1. As shown in Fig. 2(a). we construct an academic graph. It consists of multiple types of objects(Author, Paper, Venue) and relations(publish relation between papers and venues, written relation between papers and authors).



(a) Academic Network

(b) A paper path and the next chosen node

(c) Recommended paper path

Fig. 2. Result of a path-based paper recommendation.

Definition 2. *(Academic Paper Path). An academic paper path is defined as a path in the form of $v_{i_1} \rightarrow v_{i_2} \rightarrow ... \rightarrow v_{i_k}$, which describes a chain from new papers to old papers. For adjacent nodes $v_{i_{k-1}}$ and v_{i_k} in the path, there is a citation relation between them.*

Example 2. From Fig. 2(b), we can observe there is a paper path T which can be denoted as $T = v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4$. To extend the path, papers cited by v_4 will be considered and one of them will be chosen as next node according to the similarity between cited papers and T.

Definition 3. *(Path Recommendation Probability). Given a path $T = v_{i_1} \rightarrow v_{i_2} \rightarrow ... \rightarrow v_{i_k}$, which denotes a path of recommended papers, the probability of recommending the paper path is defined as the similarity between the representation of the given paper and that of the recommended path.*

Example 3. According to Fig. 2(c), there are three paper paths named $path1$, $path2$, $path3$. Then, we can return the path having the maximum similarity with v_1, after computing the similarity between v_1 and $path1, path2, path3$ separately.

Problem Formalization. Given a graph G and a query paper v_{i_1}, the Path-based Academic Paper Recommendation is to select a path T defined on the graph G, which has the maximum path recommendation probability.

3 Proposed Approach

To generate an ordered path for paper recommendation, we define the relatedness measurement of papers and the construction of path recommendation. We give an overview of our recommendation framework in Fig. 3.

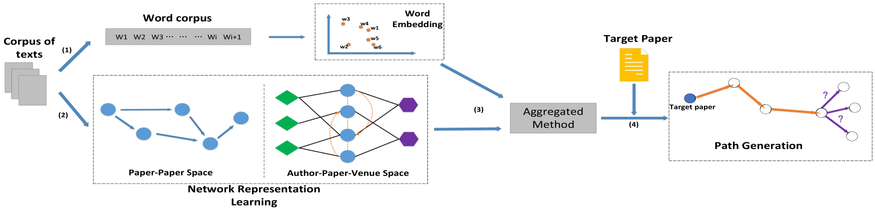


Fig. 3. Overview of recommendation framework.

3.1 Text Learning

The semantic relatedness of papers is measured mainly based on textual information. We use words from metadata (e.g., title, abstract) to build texts of documents. An effective method to compute semantic similarity of two papers is to generate word embeddings and combine them to form paper embeddings. Specifically, let c_i, c_j denote the vector representation of two papers P_i, P_j and the semantic relatedness $F_s(P_i, P_j)$ is defined by cosine function:

$$F_s(P_i, P_j) = cos(c_i, c_j) = \frac{c_i \cdot c_j}{\parallel c_i \parallel \parallel c_j \parallel} \tag{1}$$

where c_i, c_j are weighted average vector representation of words in P_i, P_j respectively. For the vector representation c_i of paper P_i, let $w = (w_1, ..., w_p)$ denote word vectors of unique words in paper and we have

$$\widehat{c_i} = \sum_{k=1}^{p} \mathrm{tfidf}(w_k) \cdot w_k \tag{2}$$

where TF-IDF is used to measure importance of word vectors and we use L2-normalization to get final text-level representation, $\mathbf{c_i} = \widehat{\mathbf{c_i}} / \parallel \widehat{\mathbf{c_i}} \parallel$.

3.2 Network Learning

The academic network is fundamentally a multi-relational heterogeneous graph where edges indicate many relationships and contain different semantic relatedness. In our model, we consider paper-paper network and author-paper-venue network and adopt two different methods to obtain relatedness between papers.

Paper-Paper Space. The paper-paper relationship constructs a homogeneous network for the academic graph. To evaluate the latent interactions between nodes, we use random walk based sampling strategy to learn latent representation of nodes. In the generation of random walks, we use two parameters p and q proposed by [5] to bias our random walks towards local neighborhood or tend to move further away. After generating a random walk, we define a window size k and $N_s(u)$ denotes neighborhood for node u in the slide window. Let $\phi_1 : V \rightarrow \mathbb{R}^d$ be the mapping function from nodes to feature representations. We now try to maximize the likelihood function

$$\max \sum_{u \in V} logPr(N_s(u)|u) \tag{3}$$

and we have

$$Pr(N_s(u)|u) = \prod_{n_i \in N_s(u)} Pr(n_i|u) = \prod_{n_i \in N_s(u)} \frac{\exp(\phi_1(n_i) \cdot \phi_1(u))}{\sum_{v \in V} \exp(\phi_1(v) \cdot \phi_1(u))} \tag{4}$$

Finally, we optimize above equation using stochastic gradient ascent and skip-gram architecture to get node representation. Then we can get the similarity of two nodes v_i, v_j in paper-paper space:

$$f_c(v_i, v_j) = cos(\phi_1(v_i), \phi_1(v_j)) = \frac{\phi_1(v_i) \cdot \phi_1(v_j)}{\parallel \phi_1(v_i) \parallel \parallel \phi_1(v_j) \parallel} \tag{5}$$

Author-Paper-Venue Space. The paper-paper relation can not discover relatedness information between papers without citation relationship. To get a better relatedness evaluation, we also consider other relationships. Specifically, we use two most common and effective meta-path schemes, which are "author-paper-author"(APA) and "author-paper-venue-paper-author"(APVPA). We use meta-path-based random walk strategy [4] to incorporate different types of nodes into skip-gram and learn effective node representations. Given a node v, we maximize the probability of having the heterogeneous context $N_t(v), t \in T_V$:

$$argmax_\theta \sum_{v \in V} \sum_{t \in T_V} \sum_{c_t \in N_t(v)} \log p(c_t|v;\theta) \tag{6}$$

where $N_t(v)$ denotes v's neighborhood with the t^{th} type of nodes. Let $\phi_2 : V \rightarrow \mathbb{R}^d$ be the mapping function for nodes and $p(c_t|v;\theta)$ is commonly defined as a softmax function:

$$p(c_t|v;\theta) = \frac{\exp^{\phi_2(c_t) \cdot \phi_2(v)}}{\sum_{u \in V} \exp^{\phi_2(u) \cdot \phi_2(v)}} \tag{7}$$

To achieve efficient optimization, we also use negative sampling for network learning. After learning network representation, the similarity of two nodes v_i, v_j in author-paper-venue space can be defined as:

$$f_a(v_i, v_j) = cos(\phi_2(v_i), \phi_2(v_j)) = \frac{\phi_2(v_i) \cdot \phi_2(v_j)}{\parallel \phi_2(v_i) \parallel \parallel \phi_2(v_j) \parallel} \tag{8}$$

Network-Based Relatedness. To make full use of network information, the final network embedding for a paper consists of paper-paper and author-paper-venue space. Formally, we formulate the relatedness of two paper v_i and v_j:

$$F_n(v_i, v_j) = \alpha f_a(v_i, v_j) + (1 - \alpha) f_c(v_i, v_j) \tag{9}$$

where $\alpha \in [0, 1]$ is to adjust the weights of two parts.

3.3 Aggregated Method

The united relatedness measurement includes semantic similarity and network closeness. We can get the relatedness $Aggr(P_i, P_j)$ between papers P_i and P_j:

$$Aggr(P_i, P_j) = \lambda F_s(P_i, P_j) + (1 - \lambda) F_n(P_i, P_j) \tag{10}$$

where $\lambda \in [0, 1]$ trades off the weight of F_s against F_n.

Time Decay. To get a path which can illustrate the development of the field of study, it is necessary to avoid overemphasizing old important articles. In this paper, we use a time-based decay parameter γ to reduce influence of older articles. The final relatedness $Sim(P_i, P_j)$ can be defined as:

$$Sim(P_i, P_j) = Aggr(P_i, P_j) \cdot \exp^{-\gamma \cdot |Time(P_i) - Time(P_j)|} \tag{11}$$

where γ is positive, $Time(P_i)$ and $Time(P_j)$ denote papers' published years.

3.4 Path Generation

To get an ordered path, we define a strategy to extend the path from a given node. We propose a two-stage beam search component for path generation, as shown in Fig. 4.

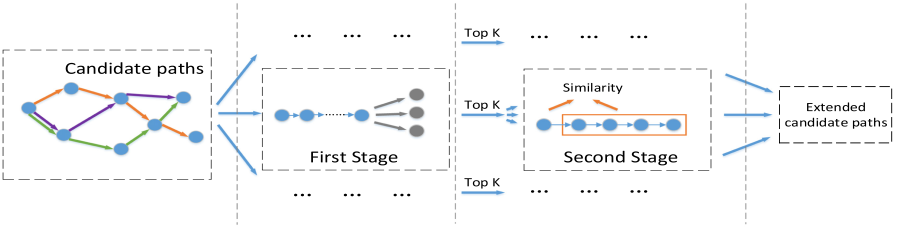


Fig. 4. Two stage path generation.

First Stage. In the first stage, we improve cohesiveness among nodes in the generated path. Specifically, given a path T in the candidate paths, which is denoted as $(u_1, u_2, ..., u_l)$, we need to choose the next node to extend the path. The similarity of candidate node $\widehat{u}$ and path T can be defined as:

$$F_h(T, \widehat{u}) = \frac{1}{l} \sum_{i=1}^{l} Sim(u_i, \widehat{u}) \tag{12}$$

To get better results, we use beam search to get a set of candidate nodes $CanSet = \{u_{l+1}^{(1)}, ..., u_{l+1}^{(k)}\}$, which has top-$k$ high similarity with the path. We extend the given path T to k new paths. For all candidate paths, we do the same operation and get the new set of all extended candidate paths.

Second Stage. In the second stage, we try to reduce the size of the set of extended candidate paths. To keep topic similarity between the generated paths and the original paper, we define the similarity between the path $T = (u_1, u_2, ..., u_l)$ and the original node u_1 as follows:

$$F_h(T, u_1) = \frac{1}{l-1} \sum_{i=2}^{l} Sim(u_i, u_1) \tag{13}$$

We choose the most similar k candidate paths as the i-step generated paths. We do two-stage path-generation repeatedly to get the final generated paths and rank these paths in order of decreasing similarity for paper recommendation.

4 Experiment

In this section, we introduce the datasets and experimental setup, as well as the evaluation metrics and baseline methods. We also present and analyze the results of our experiments.

4.1 Data Preparation

Dataset. The experiments are conducted on two different datasets: the DBLP dataset and the ACM dataset [19]. The information of title, abstract, keywords, authors, publication venue and publication year is used in both datasets. Statistics of the two constructed heterogeneous bibliographic networks are summarized in Table 2. We preprocess texts of each paper, remove stop words and words appearing less than 10 times and then stem each word.

Table 2. Statistics of two datasets.

Dataset	Papers	Authors	Venues	Terms	Relationships
DBLP	1,782,700	2,052,414	18,936	100,000	9,590,600
ACM	2,385,057	2,004,398	269,467	61,618	12,048,682

Evaluation Settings. To conduct the experiments, we invite 10 experts to help on evaluating the quality of recommended results. We randomly choose 200 papers in DBLP and ACM as test set respectively. For each given paper, 3 experts annotate ordered paths and we choose the overlap results as ground truth papers. All meta-data (e.g., title and abstract) can be available for annotators.

4.2 Experimental Settings

Baseline Methods. Several widely deployed paper recommendation approaches were implemented. We compared the recommendation results of the following methods in academic network:

LDA: LDA [1] is a celebrated generative model for text documents that learns representations for documents as distributions over word topics.

LSI: LSI [3] uses singular value decomposition on the BOW representation to arrive at a semantic feature space.

PathSim: PathSim [17] is a meta path-based method to search similar papers in heterogeneous information networks. It considers different linkage paths to study similarity among the same type of objects in academic networks.

HeteSim: HeteSim [15] is a path-constrained method to measure the relatedness of heterogeneous objects in heterogeneous networks.

PageRank: PageRank [10] is a method to derive an object's importance based on authority propagation in the heterogeneous bibliographic network. It tends to rank papers based on citation analysis in citation networks.

Deepwalk: Deepwalk [12] is a network-only representation learning method. Deepwalk takes random walk paths from network as sentences and nodes as words to learn the node representations by applying the Skip-Gram algorithm.

Node2vec: Node2vec [5] proposes an improvement to the random walk phase of DeepWalk and combines DFS-like and BFS-like neighborhood exploration.

PAPR: Our proposed method combines text and network structure information to learn representations for nodes. To evaluate the benefit brought by time decay introduced in Sect. 3.3 and two-stage path-generation introduced in Sect. 3.4, we design two variations of PAPR. **PAPR1** only uses two-stage path-generation and **PAPR2** only uses time decay.

Parameter Settings. In the section of text learning, we adopt 100 dimension, 10 window size, skip-gram model and 5 negative samples for word2vec. In the section of network learning, we set 128 dimension, 10 window size as the basic parameters. The number of walks per node is 20 and the walk length is 30. In all methods, biased parameters are also fine-tuned to report the best performance.

Table 3. Recommendation performance comparisons on DBLP and ACM datasets in terms of Precision and Recall.

Dataset	DBLP						ACM					
	P@3	P@5	P@8	R@3	R@5	R@8	P@3	P@5	P@8	R@3	R@5	R@8
LDA	0.3686	0.2725	0.1873	0.1382	0.1538	0.1586	0.2923	0.2128	0.1726	0.1182	0.1287	0.1578
LSI	0.3813	0.2796	0.2043	0.1419	0.1564	0.1715	0.3769	0.2615	0.1794	0.1516	0.1575	0.1625
PathSim	0.3729	0.2599	0.1761	0.1402	0.1469	0.1497	0.3684	0.2519	0.1716	0.1476	0.1513	0.1540
HeteSim	0.3665	0.2556	0.1760	0.1383	0.1450	0.1505	0.3549	0.2529	0.1857	0.1195	0.1273	0.1415
PageRank	0.3841	0.2848	0.2133	0.1410	0.1568	0.1761	0.4211	0.2895	0.1934	0.1654	0.1700	0.1705
Deepwalk	0.4067	0.2966	0.2071	0.1526	0.1677	0.1761	0.3730	0.2538	0.1692	0.1509	0.1541	0.1542
Node2vec	0.4258	0.3220	0.2222	0.1589	0.1807	0.1872	0.4576	0.3179	0.2119	0.1837	0.1911	0.1931
PAPR1	0.4892	0.3714	0.2762	0.1807	0.2060	0.2308	0.5423	0.4153	0.3042	0.2187	0.2507	0.2756
PAPR2	0.5084	0.3771	0.2777	0.1895	0.2109	0.2330	0.4346	0.3179	0.2154	0.1758	0.1927	0.1957
PAPR	**0.5443**	**0.4218**	**0.3325**	**0.2023**	**0.2345**	**0.2691**	**0.6231**	**0.4948**	**0.3709**	**0.2521**	**0.3001**	**0.3356**

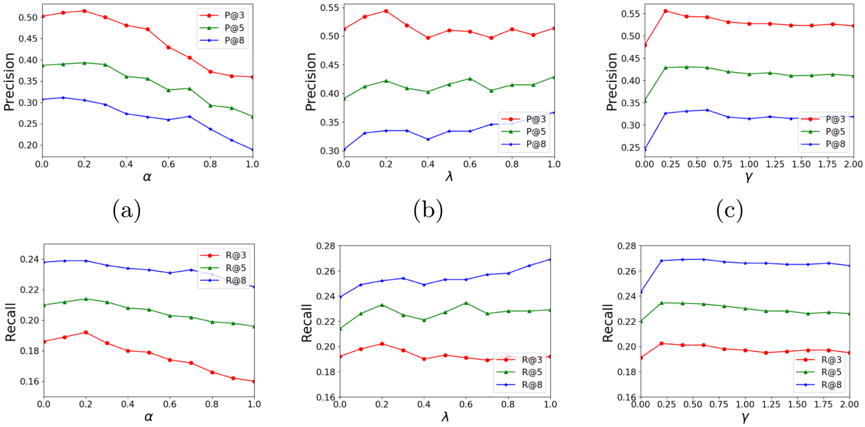
Evaluation Metrics. We employ Precision and Recall at position M (P@M and R@M) as the evaluation metrics. Precision@M is defined as the fraction of ground truth papers contained by the M-length recommended path and Recall@M is defined as the fraction of ground truth papers that appear in the M-length recommended path contained by the whole ground truth. For each given paper P_i, we have:

$$P@M = \frac{|N(M; P_i)|}{|N_p(P_i)|} \qquad R@M = \frac{|N(M; P_i)|}{|N_r(P_i)|} \tag{14}$$

where $N(M; P_i)$ is the set of ground truth papers in the M-length recommended path, $N_p(P_i)$ is the set of papers in the recommended path and $N_r(P_i)$ is the set of papers in the ground truth path.

4.3 Result Analysis

Evaluation results on different methods are shown in Table 3. We compare the proposed method with different baselines using Precision and Recall. Table 3 summarizes the comparison results on both DBLP and ACM datasets. It can be easily observed that our proposed method outperforms other methods and gains a performance improvement of more than 20% over the best competitive algorithm. In general, the method LDA gets the lowest precision on ACM dataset, which shows that the similarity of papers can not be evaluated only based on content information. PathSim and HeteSim have the similar performance. However, they don't perform well on DBLP dataset because they heavily depend on hand-engineered meta paths. To analyze the effect of time-decay component and two-stage path generation component in our method, we also compare the proposed method with its variation PAPR1 and PAPR2. We can observe that PAPR gains a performance improvement of more than 10% over PAPR1 and more than 6% over PAPR2 on precision metrics, since time-decay strategy and two-stage path-generation play important roles for better recommendation.



(a) (b) (c)

(d) (e) (f)

Fig. 5. Precision and recall with varying α, λ and γ for PAPR.

4.4 Parameter Study

In this section, we study the impact of parameters α, λ and γ on dataset DBLP. α is proposed to balance paper-paper space and author-paper-venue space. Observed from Fig. 5(a)(d), PAPR achieves the best performance when $\alpha = 0.2$. This is because, citation relation has a leading role and two meta-path relations (APA, APVPA) make a great supplement for relatedness measurement. λ trades off the weight of text semantic and network structure. We set $\alpha = 0.2$ to evaluate influence of text similarity and network similarity. From Fig. 5(b)(e), we find that the lower λ can get better results of Precision@3 and Recall@3. Meanwhile, the higher λ gets better results of Precision@8 and Recall@8. It means that short path depends more on network structure and long path depends more on text semantics. γ is used to reduce the influence of older papers. Figure 5(c)(f) show the influence of time decay. We can observe that the precision and recall increase quickly at first and then decline slowly, which means time decay is helpful to get a better time-oriented path but excessive decay has no better effects.

To achieve the trade-off between running time and the quality of the paths, we also perform experiments over average running time of different beam sizes

Table 4. Performance comparision over running time of different beam sizes.

Beam size	DBLP							ACM						
	Time(s)	P@3	P@5	P@8	R@3	R@5	R@8	Time(s)	P@3	P@5	P@8	R@3	R@5	R@8
1	0.323	0.510	0.395	0.296	0.172	0.209	0.237	0.410	0.518	0.439	0.356	0.229	0.267	0.311
3	0.662	0.526	0.413	0.328	0.181	0.222	0.256	0.695	0.579	0.473	0.365	0.234	0.277	0.328
5	0.935	0.545	0.421	0.333	0.202	0.234	0.269	1.020	0.623	0.495	0.371	0.252	0.300	0.335
7	1.194	0.551	0.424	0.335	0.213	0.235	0.273	1.371	0.617	0.494	0.373	0.249	0.300	0.336

on each testcase. Table 4 illustrates the results w.r.t. Precision and Recall on different beam sizes in DBLP and ACM dataset. We can see that both running time and quality of paths increase as the beam size increases. For running time, it increases steadily as beam size increases. However, there is a much more rapid increase on performance when beam size increases from 1 to 5, and Precision and Recall have few improvements from 5 to 7 in both datasets. Therefore, we set beam size as 5 to obtain most of the benefits of efficiency and performance.

5 Related Work

5.1 Content-Based Algorithms

Content-based filtering is a widely used method to compare similarity of items in paper recommendation systems. It uses keywords extracted from papers' texts to evaluate similarity of articles. [21] feeds documents' titles and abstracts into TF-IDF model and learns probabilistic model to evaluate relatedness of documents. [9] develops a paper recommendation system which constructs users' profiles based on candidate papers' titles. However, these methods mostly use Bag-of-Words model, which has difficulty in finding conceptually similar work [2].

5.2 Collaborative Filtering

Collaborative filtering is a popular and widely used method in recommender systems. However, collaborative filtering cannot generate accurate recommendations without sufficient initial ratings from users. In order to alleviate the cold start problem, [8] proposes a context-based collaborative filtering, which incorporates co-occurrence relations into rating matrix. [14] combines the matrix factorization with the topic modeling to achieve a better performance. Nevertheless, collaborative filtering needs much computing time and offline data processing.

5.3 Network-Based Algorithms

Network-based paper recommendations concentrate on analyzing citation network to understand the relationship between scholarly papers. Two main methods in citation analysis are co-citation analysis and bibliographic coupling. However, these methods can not address complex relationships in networks [22].

[20] proposes a method based on random walk with restart(RWR) to measure vertex-to-vertex relevance. [7] considers a heterogeneous network to recommend similar papers. Nevertheless, these works mainly extract hand-engineered structural information to calculate paper similarity, which is inflexible and time-consuming.

6 Conclusion

In this paper, we propose a Path-based Academic Paper Recommendation, namely PAPR, to perform academic paper recommendation. Specifically, to acquire an ordered path of relevant scholarly papers, we design a flexible and expressive model, which takes both text semantics and network structure into account. To facilitate evaluation of similarity between papers, textual attributes and network structure are embedded in vector space. The experiments based on real datasets show that our model outperforms other methods.

Acknowledgment. This work was supported by the Major Program of the Natural Science Foundation of Jiangsu Higher Education Institutions of China under Grant No. 19KJA610002 and 19KJB520050, and the National Natural Science Foundation of China under Grant No. 61902270, a project funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
2. Chen, T.T., Lee, M.: Research paper recommender systems on big scholarly data. In: Yoshida, K., Lee, M. (eds.) PKAW 2018. LNCS (LNAI), vol. 11016, pp. 251–260. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-97289-3_20
3. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inform. Sci. **41**(6), 391–407 (1990)
4. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 135–144 (2017)
5. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864 (2016)
6. Hammou, B.A., Lahcen, A.A., Mouline, S.: APRA: an approximate parallel recommendation algorithm for big data. Knowl.-Based Syst. **157**, 10–19 (2018)
7. Lao, N., Cohen, W.W.: Relational retrieval using a combination of path-constrained random walks. Mach. Learn. **81**(1), 53–67 (2010)
8. Liu, H., Kong, X., Bai, X., Wang, W., Bekele, T.M., Xia, F.: Context-based collaborative filtering for citation recommendation. IEEE Access **3**, 1695–1703 (2015)
9. Nascimento, C., Laender, A.H., da Silva, A.S., Gonçalves, M.A.: A source independent framework for research paper recommendation. In: Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries, pp. 297–306 (2011)

10. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: bringing order to the web. Proceedings of the WWW Conference, vol. 1998, pp. 161–172 (1999)
11. Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_10
12. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710 (2014)
13. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 291–324. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_9
14. Shan, H., Banerjee, A.: Generalized probabilistic matrix factorizations for collaborative filtering. In: 2010 IEEE International Conference on Data Mining, pp. 1025–1030. IEEE (2010)
15. Shi, C., Kong, X., Huang, Y., Philip, S.Y., Wu, B.: HeteSim: a general framework for relevance measure in heterogeneous networks. IEEE Trans. Knowl. Data Eng. **26**(10), 2479–2492 (2014)
16. Small, H.: Co-citation in the scientific literature: a new measure of the relationship between two documents. J. Am. Soc. Inform. Sci. **24**(4), 265–269 (1973)
17. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: PathSim: meta path-based top-K similarity search in heterogeneous information networks, vol. 4, pp. 992–1003. CiteSeer (2011)
18. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, pp. 3104–3112 (2014)
19. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: 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 (2008)
20. Tian, G., Jing, L.: Recommending scientific articles using bi-relational graph-based iterative RWR. In: Proceedings of the 7th ACM Conference on Recommender Systems, pp. 399–402 (2013)
21. Wang, C., Blei, D.M.: 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 (2011)
22. Zhou, D., et al.: Learning multiple graphs for document recommendations. In: Proceedings of the 17th International Conference on World Wide Web, pp. 141–150 (2008)