Author Contributed Representation for Scholarly Network

Binglei Wang, Tong Xu(✉), Hao Wang, Yanmin Chen, Le Zhang, Lintao Fang, Guiquan Liu, and Enhong Chen

School of Computer Science and Technology, University of Science and Technology of China, Hefei, China
blwang59@gmail.com, tongxu@ustc.edu.cn,
{wanghao3, ymchen16}@mail.ustc.edu.cn, zhangle0202@gmail.com,
flt@mail.ustc.edu.cn, {gqliu,chenen}@ustc.edu.cn

Abstract. Scholarly network analysis is a fundamental topic in academia domain, which is beneficial for estimating the contribution of researchers and the quality of academic outputs. Recently, a popular fashion takes advantage of network embedding techniques, which aims to learn the scholarly information into vectorial representations for the task. Though great progress has been made, existing studies only consider the text information of papers for scholarly network representation, while ignoring the effects of many intrinsic and informative features, especially the different influences and contribution of authors and cooperations. In order to alleviate this problem, in this paper, we propose a novel Author Contributed Representation for Scholarly Network (ACR-SN) framework to learn the unique representation for scholarly networks, which characterizes the different authors' contribution. Specifically, we first adopt a graph convolutional network (GCN) to capture the structure information in the citation network. Then, we calculate the correlations between authors and each paper, and aggregate each embedding of authors according to their contribution by using the attention mechanism. Extensive experiments on two real world datasets demonstrate the effectiveness of ACR-SN and reveal that authors' contribution to the paper varies with the corresponding authorities and interested fields.

Keywords: Scholarly network embedding · Scholar cooperation · Graph convolutional network

1 Introduction

Recent years have witnessed the rapid accumulation of scholarly data, containing rich information of research publishing records with citation networks, which provides unprecedented opportunities for scholarly network analysis [25]. Indeed, with the help of scholarly network analysis, on one hand, we could uncover the trend of research. On the other hand, it is convenient for researchers to choose an appropriate partner and evaluate the influence of work from the micro view.

X. Wang et al. (Eds.): APWeb-WAIM 2020, LNCS 12317, pp. 558–573, 2020.
https://doi.org/10.1007/978-3-030-60259-8_41

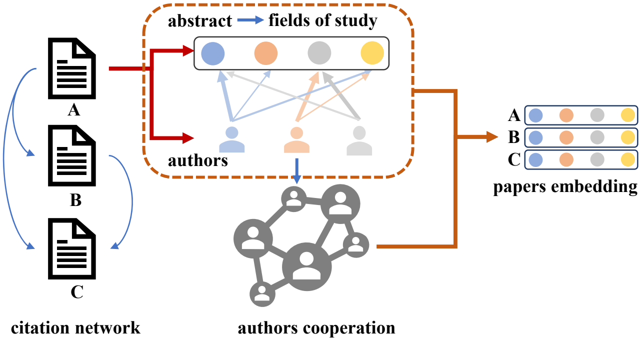

Fig. 1. The overview of scholarly network embedding.

Towards scholarly network analysis, there are many kinds of studies, such as predicting the authorities of authors [12], predicting the influence of paper [4] and paper recommendation [24]. Though large efforts have been made, the researchers usually consider the paper textual information, while the great benefits of academic cooperation are largely under-explored. In fact, cooperation is of great importance for scholars, especially for young researchers. Therefore, in this paper, we aim to study a more comprehensive scholarly network analysis by considering cooperation effects.

In academic networks, dissertations are often considered as research units that can be clustered to higher levels by domains or authors [26]. As shown in Fig. 1, three papers (i.e., A, B, C) can form a citation network, e.g., paper A cites paper B and C. We can make deep analysis about their abstract contents and authors. Specifically, the abstracts generally reflect their study fields and the authors can constitute a co-author relationship which demonstrates the authority of each researcher in different fields. Moreover, different authors may make different contribution to a paper, due to their various authorities and areas of interest. Collaboration in each paper can be obtained by summarizing the embedded vector of each author. By combining these aspects, the preliminary performance of the dissertations is fully formed. In addition, the cited neighbours of the paper are often in closer research fields in the citation network. Therefore, this constraint on similarity should be retained when learning the scholarly network.

Along this line, we propose a novel scholarly network embedding framework called Author Contributed Representation for Scholarly Network (ACR-SN). We first extract the study fields from paper abstract and embed authors of the papers. Then we combine authors embedding and study fields in the paper through the novel author-paper attention mechanism, which could capture the authors' influences and interests. Considering the different contribution of authors to a certain paper, the contribution attention layer is introduced to form the initial representation of papers from the aggregation of authors embedding. Next, we utilize graph convolutional network (GCN) to preserve the citation based similarities of neighbors and the structural features in citation network. Finally, extensive experiments on several scholarly networks demonstrate the effectiveness of our model.

In summary, the major contribution of this paper can be briefly summarized as follows:

– We propose a novel framework (ACR-SN), which describes the different influences and interests of authors for each paper.
– We adopt two layers of attention network. The first is to catch the influence of authors in the paper, and the second is to measure different contribution of authors which leads to better initial embedding of the paper. Considering the similarities in paper and its references, we use GCN to incorporate the paper attributes in information diffusion of the network.
– We conduct extensive experiments on two real world datasets, which demonstrate our ACR-SN framework shows significant performance in many downstream tasks including paper classification and citation prediction.

2 Related Work

In this section, we will summarize the related works in scholarly data analysis and network embedding techniques.

Scholarly Data Analysis. Scholarly data contains multiple scholarly entities, e.g., papers and authors, as well as multiple scholarly relations, e.g., citations among papers, co-authors relationship among authors [25]. Among different scholarly networks, there are various analysis and applications. As for the citation network, research [4] predicts the influence of paper, and research [24] recommends paper based on citation and hierarchical structure of scientific knowledge. For a more comprehensible way of research articles organization, some researchers form a study map [20]. As for the co-author network, some researchers predict the influence and authority of authors using cooperation information [12]. Some studies also analyze authors' contribution with different relations among them [17]. Among these various analysis in scholarly network, the embedding of scholarly entities, authors and papers both are the fundamental issues to solve. This paper focuses on citation networks, and the research object is paper.

Network Embedding. Network embedding is intensively studied these years. The aim of network embedding is to get a low dimensional representation which can model the structure and some other properties of network. There are mainly three kinds of methods:

The first kind of methods are based on matrix-factorization, for example, the well-known Laplacian eigenmaps (LE) [3] and graph factorization (GF) [1]. These methods utilize the eigenvectors as the network representation.

The second kind is based on random walk. These methods use truncated random walk to get the neighbors representation of nodes to decrease the complexity. DeepWalk [16] and node2vec [7], as two typical methods, are also based on inner product of node pairs. However, unlike the matrix-factorization methods, these methods learn nodes embedding to maximize the probability of visiting two nodes on one truncated random walk, rather than using a deterministic node similarity measure.

The third kind of embedding methods combine node attributes and network structure. The previous two methods learn the node represetation from the structure of network, while node attributes are ignored. Unlike them, TADW [27] is based on deepwalk while incorporateing node information. In scholarly network embedding field, Paper2vec [6] combines graph and text information of paper to form the representation. Except for the supervised representations, there are some unsupervised methods, such as UPPSNE [28], and SANE [22], which use pairwise node embedding to represent node; MCNE [23], which learns multiple preference of users in the social network. Also there are some task specific methods, for example, LSNE [5] is a link-oriented signed network embedding method, and DLPQV [11] uses network embedding method to evaluate the quality of patents. Furthermore, some researchers use deep learning methods, which expand the convolution from Euclidean domain to non-Euclidean domain, and these methods are called graph convolutional networks. Among these methods, GCN [9] uses the first-order neighbors to simplify the filter in convolutional network. To get representation inductively, GraphSAGE [8] learns the aggregation of a node's neighbor, instead of learning a deterministic node embedding. GAT combines attention mechanism into graph convolutional network, considering different influence of nodes' neighbours. Additionally, there is some improvement methods like Geom-GCN [15], which proposes geometric aggregation scheme for graph neural networks to overcome the weakness of message-passing neural networks used in GCN. The deep learning based embedding methods inspire us to use GCN to represent articles in citation network.

3 Preliminary and Problem Definition

In this section, we give the definition to the scholarly network embedding problem. To get a better embedding of paper, here we use both author cooperation and text information of paper to represent it. Let p denote a paper from the corpus P. For the information in paper, we use $x_p \in 1 \times d$ to denote the abstract text of p, which consists of the averaged d-dimensional words embeddings of words in the abstract. As for the citations among papers, this relationship can be represented by an adjacency matrix $C \in \mathbb{R}^{|P| \times |P|}$, where $c_{ij} \in \{0, 1\}$ denotes if there is citation relationship between paper p_i and paper p_j. There is also a set of authors Au of the research papers, and each paper p corresponds to an author group au_p, which is a subset of Au.

Given the preliminaries above, we define the problem to solve in this paper:

Table 1. Summary of notations.

Notation	Definition
G	Graph
P	Set of papers in scholarly network
C	Adjacency matrix of paper citations in scholarly network
Au	Set of paper authors
au_i	Authors group of paper i
$\boldsymbol{X}$	Text embedding of paper
$\boldsymbol{x}_i$	Text embedding of paper i,$l \times d$
d	Dimension of embeddings
$\boldsymbol{V}$	Embedding of the papers in the network,$\|P\| \times d$
$\boldsymbol{v}_i$	Embedding of the paper i,$1 \times d$
$\Theta^{(i)}$	Parameters in layer i of GCN
k	Number of paper classes
Z	Paper classification prediction, $\|P\| \times k$
l	Number of words in paper p
$\boldsymbol{author}_j$	Embedding of author j,$1 \times d$
ac_{ij}	Importance of author j to the paper i,$1 \times d$
$nhid$	Number of hidden layers in GCN
$\odot$	Element-wise multiplication

Definition 1. ***Scholarly Network:*** *A scholarly network would be denoted as $G = (P, C, Au, \boldsymbol{X})$, where P is a set of paper, C is a set of citations and references among these papers. $\boldsymbol{X}$ is the text information in paper, here is the average word embeddings of each paper's abstract.*

Definition 2. ***Scholarly Network Embedding:*** *Given a scholarly network $G = (P, C, Au, \boldsymbol{X})$, the aim of scholarly network representation is to get a representation $\boldsymbol{v}_i$ of each paper p_i in a low-dimension space, combining the information of paper and the corresponding authors. The target is to lessen the classification loss between the categories predicted using representation $\boldsymbol{v}_i$ and true labels.*

4 Author Contributed Representation for Scholarly Network Framework

In this section, we propose a model ACR-SN to represent paper in scholarly network, the framework of which is shown in Fig. 2. The whole structure of our model consists of three parts: 1) Paper information input; 2) Author-abstract pairwise attention and author contribution attention layer, which is to fully capture the influence of the authors and learn the attribute of paper from its

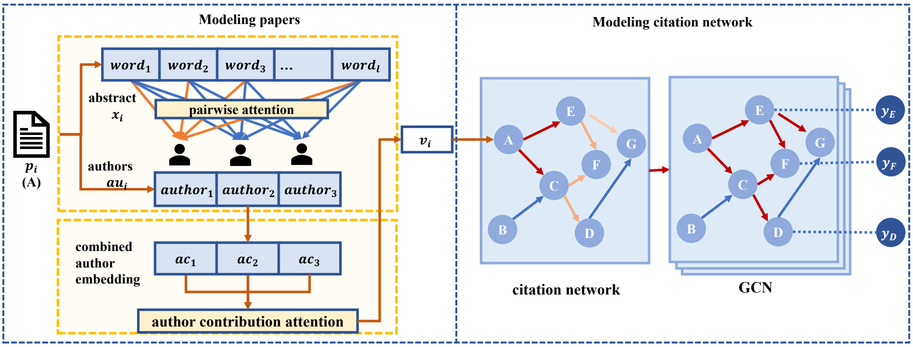

Fig. 2. Framework of Author Contributed Representation for Scholarly Network (ACR-SN). The left part is papers modeling and right is citation network modeling.

abstract; 3) graph convolutional network which is utilized to preserve the structure and transmit the node embeddings in the citation network. The notations are shown in Table 1.

4.1 Network Input

The input of ACR-SN is a citation network, in which each node is a paper, containing authors and abstracts. Take paper i as an example, author j in author group au_i author is mapped to an embedding vector $author_j$ using one hot embedding, word k in the abstracts also reflect to a same dimensional vector $word_k$ by word2vec embedding method. To construct embedding for each paper with the author and abstract information, our model is introduced as below.

4.2 Modeling Papers

Paper modeling is the core part of our method, which aims to capture author and abstract information. It consists of two layers of mechanism. The first layer is a pairwise attention between authors and study areas of the paper, which captures the author's expertise and interest in the areas covered by the paper to generate the embeddings of authors in each paper. The second layer is used to captures the different contribution of authors to the same paper.

Paper-Author Pairwise Attention. The abstract of paper i could be split into several topics through the words embeddings $word_k$. Meanwhile, the authors in au_i also appear in multiple papers, indicating that they have different research interests. To emphasize differences in authors' interests and papers' topics, here we utilize a pairwise method to model the interaction between papers and authors. The details of this attention layer are shown below.

For paper i and author j in au_i, the representation of paper i is calculated as follows:

$$\boldsymbol{ac}_{ij} = Mean(\boldsymbol{x}_i \odot \boldsymbol{author}_j), \tag{1}$$

$$\boldsymbol{ac}_i = \left[\boldsymbol{ac}_{i1}, \ldots, \boldsymbol{ac}_{ij}, \ldots, \boldsymbol{ac}_{|au_i|}\right]. \tag{2}$$

Here $\boldsymbol{x}_i \in \mathbb{R}^{l \times d}$ is the paper embedding matrix, which is the average of lookup vectors of the words in the abstract of paper i, and l is the total number of words. $\boldsymbol{author}_j$ is the embedding of author j in au_i. $\odot$ represents the elementwise product between these vectors. Notice that $\boldsymbol{author}_j \in \mathbb{R}^d$ would be filled to $\mathbb{R}^{l \times d}$ automatically, and $\boldsymbol{ac}_{ij} \in \mathbb{R}^{1 \times d}$ is column average of the elementwise product, representing the embedding of author j in paper i. After column concatenation, the aggregated paper representation vector is $\boldsymbol{ac}_i \in \mathbb{R}^{|au_i| \times d}$.

After generating the authors embeddings, we use another attention layer to aggregate them to form the embedding of paper. The second attention layer is the author attention, which is discussed in the next subsection.

Author Attention. To get the paper representation, an intuitive idea is to stack the representation of authors together and use the average pooling to get paper representation vector. However, this idea ignores the fact that each author may contributes to the paper differently. So we introduce the attention mechanism to apply different importance to each author. The attention layer is a linear layer to learn each author's contribution $\boldsymbol{a}_i$ to the paper i. The detailed attention weight learning progress is shown below:

$$\boldsymbol{a}'_i = W \cdot \boldsymbol{ac}_i + b, \tag{3}$$

where $\boldsymbol{ac}_i \in \mathbb{R}^{|au_i| \times d}$ is the vectors of the authors' representations of paper i. $A_i \in \mathbb{R}^{1 \times d}$ is the attention weight of the authors to the paper i, and b is the bias vector. The output $\boldsymbol{a}'_i \in \mathbb{R}^{|au_i|}$ is the attention weights of authors of paper i. The attention weight $\boldsymbol{a}_i$ is normalized by the softmax function:

$$\boldsymbol{a}_i = \frac{exp(a'_{ij})}{\sum_{j=1}^{|au_i|} exp(a'_{ij})}. \tag{4}$$

Here a'_{ij} is the j-th component of $\boldsymbol{a}'_i$. The normalization makes sure each author's attention weight is in $[0, 1]$, and the sum of authors contribution is 1.

The paper representation $\boldsymbol{v}_i$ is calculated in the following form:

$$\boldsymbol{v}_i = \boldsymbol{ac}_i \cdot \boldsymbol{a}_i. \tag{5}$$

In this equation, $\boldsymbol{ac}_i$ and $\boldsymbol{a}_i$ are calculated in Eq. (2) and Eq. (4), and the output $\boldsymbol{v}_i \in \mathbb{R}^d$ is the representation vector of paper i. After the calculation in this subsection, the initial embedding of articles is formed. By using two layers of attention in authors and abstracts, we incorporate different kinds of information in scholarly data. To combine the citation structure and learn an accurate representation, we will introduce the GCN framework in the next section.

4.3 Modeling Citation Network

In this part we will show how to use GCN to form the final representation of paper i under the constraint of similarities in its citations.

Here all papers in the dataset P compose a feature matrix $V \in |P| \times d$, where $|P|$ is the number of papers in the citation network, d is dimension of feature vectors. The citation network can be presented in adjacency matrix C, which is generated from paper set P and citations in this network; the degree matrix is denoted as D. Following the spectral approaches in graph neural network, GCN limits the convolution operation to one-localized to avoid overfitting, and uses renormalization trick to refrain from numerical instabilities and exploding or vanishing gradients. So the aggregator in GCN is $\hat{C}X$, where $\hat{C} = \tilde{D}^{-\frac{1}{2}}\tilde{C}\tilde{D}^{-\frac{1}{2}}$ which is a normalization trick in GCN [9], in which $\tilde{C} = C + I_N, \tilde{D}_{ij} = \sum_j \tilde{A}_{ij}$. The forward process is described in the following part.

The input of this part is the representation matrix of paper V, consists of vectors calculated by Eq. (5). The weight in the first layer is denoted as $\Theta^{(0)}$, and the calculation in first layer is shown as below:

$$F = \hat{C}V\Theta^{(0)}, \tag{6}$$

where $\Theta^{(0)} \in \mathbb{R}^{|P| \times nhid}$ is the weight of first layer. $nhid$ is the number of hidden layers in GCN. The output $F \in \mathbb{R}^{|P| \times nhid}$ is the input of next graph convolution layers. The structure of the next layer is similar to the first layer, except the ReLU unit and softmax layer.

$$Z = softmax(\hat{C}ReLU(F)\Theta^{(1)}), \tag{7}$$

where $Z \in \mathbb{R}^{|P| \times k}$ is the convolved signal matrix, and k is the final number of classification of papers. The softmax layer is applied row-wise. The weight in this layer is $\Theta^{(1)} \in \mathbb{R}^{nhid \times k}$. The output of the model is the probability of each type which the paper is divided into. And the next subsection will show the learning process.

4.4 Model Learning

Objective Function. For the proposed ACR-SN model, we use cross entropy shown in Eq. 8 to promise that papers are divided into correct area as much as possible. As mentioned in GCN [9], the loss function is defined in cross-entropy form to maximize the similarity of node representation to the node label:

$$L = -\sum_{l \in Y_L} \sum_{f=1}^{k} Y_{lf} ln Z_{lf}. \tag{8}$$

Here Y_L is the labeled set. k is the number of node classes. Y_{lf} is the vector of true labels, and Z_{lf} is the predicted possibilities of each paper in each class. To optimize our model, we use Adam optimizer to learn the parameters.

Parameter Initialization. In the cooperation attention part, we initialize the authors initial weight to all 1 vectors, assuming all the authors contributes equally to the paper features. For the initial author embedding, we use word embedding to get low dimension one hot embedding of authors and words. As for words in the abstract, we use word2vec to generate their initial embeddings. For all the words in the abstract in the dataset, we select the top 3000 frequent words for a brief embedding. The author and word embedding are randomly initialized and can be learned during training.

In the GCN part, we initialize the weight with a Gaussian distribution, with a mean of 0 and deviation of $1/\sqrt{out_{dimension}}$.

5 Experiments

5.1 Experimental Settings

Datasets. For the purpose of learning the embedding of papers in the scholarly network, here we conduct the experiments on two scholarly networks to demonstrate the effectiveness of our proposed model ACR-SN:

- Semantic Scholar [2]. This is an open scholarly database. Here we downloaded the 2017-10-30 version from the Semantic Scholar website. In [14], they constructed DBLP dataset by extracting four study fields, namely Database, Data Mining, Artificial Intelligence and Computer Vision. In this paper, we also used these four areas, and filtered the data in Semantic Scholar dataset to extract the paper in these fields. After the preprocessing step, there are 48,878 papers. The max connected subgraph contains 46,637 papers and 174,185 citation links.
- DBLP [18]. This is a famous paper dataset in computer science. After filtering out the papers in the four areas mentioned before, there are 78,939 papers in the dataset.

The detailed statistics of datasets is shown in Table 2. These two datasets are both popular in scholarly data mining. The number of four kinds of papers is

Table 2. The statistics of datasets.

Datasets		Semantic scholar	DBLP
#Nodes		48,878	62,137
#Links		174,622	319,222
#Authors		47,343	77,260
Study fields	DB	4,579	6,340
	DM	23,851	10,956
	AI	7,754	20,915
	CV	12,694	23,962

basically balanced, except for the relatively small ones in Database field. The quantity of authors and papers are similar in both datasets. And DBLP is slightly larger compared with Semantic Scholar dataset.

Baselines. As mentioned in the Sect. 2, we selected several state-of-the-art methods to demonstrate the effectiveness of our learned scholarly network embedding by ACR-SN:

Structure-based Methods:

- **Node2vec** [7], different from DeepWalk [16], it designs a biased truncated random walks to efficiently explore diverse neighborhood and utilizes the skip-gram model to learn the node embedding.
- **LINE** [19] is a method that defines the first-order and second-order proximity of network structure to obtain the node representation, respectively.

Combined Methods:

- **Node2vec+attr** is a method using combined features of Node2vec and paper attributes to classify the paper.
- **LINE+attr** also combines the representation of LINE with paper features.
- **UPP-SNE** [28], which is the abbreviation of user profile preserving social network embedding, learns the node embedding by preserving the structure of network and node attributes simultaneously.
- **Paper2vec** [6] solves the problem similar to our method. This method learns the embedding of paper from text information, and uses the citation network structure to jointly refine the learned embedding.

Deep Learning based Methods:

- **GraphSAGE** [8] is a general inductive network embedding framework which generates embedding by aggregating features from a node's neighbors.
- **GCN** [9] optimizes the node embedding in a semi-supervised framework, which has the similar objective function with our method.
- **GAT** [21] considers different weights of neighbors to a node in a network, using attention mechanism in graph neural network.
- **ACR-SN-avg** is the reduced version of our proposed model ACR-SN without containing the attention part.

Evaluation. In the classification experiments, the evaluation metric we used is Accuracy, which is defined by the portion between nodes classified correctly and the total number of nodes:

$$accuracy = \frac{\#nodes\ classified\ correctly}{\#nodes}. \tag{9}$$

As for the link prediction task, we used average precision(AP) and area under curve(AUC) to evaluate the effectiveness of experiments. For each experiment, we randomly selected 10% to 90% from the dataset as training set, and split the remaining part to validation set and test set.

Table 3. The experimental results of node classification on semantic scholar.

Methods	Training ratio								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Node2vec	0.7002	0.7035	0.7040	0.7063	0.7076	0.7069	0.7082	0.7068	0.7144
LINE	0.6368	0.6421	0.6436	0.6457	0.6445	0.6460	0.6461	0.6464	0.6571
Node2vec+attr	0.7344	0.7442	0.7481	0.7505	0.7513	0.7525	0.7544	0.7565	0.7627
LINE+attr	0.6883	0.7046	0.7138	0.7167	0.7115	0.7192	0.7177	0.7117	0.7308
UPP-SNE	0.6113	0.6160	0.6187	0.6192	0.6196	0.6205	0.6210	0.6267	0.6230
Paper2vec	0.6869	0.6915	0.6933	0.6969	0.6972	0.6967	0.6993	0.7003	0.7122
GraphSAGE	0.4748	0.4748	0.4877	0.4884	0.4886	0.4903	0.4885	0.4898	0.5018
GCN	0.7141	0.7193	0.7162	0.7334	0.7332	0.7387	0.7429	0.7427	0.7480
GAT	**0.7958**	**0.7982**	**0.7968**	**0.8013**	0.7997	0.7998	0.8050	0.8080	0.8130
ACR-SN-avg	0.7683	0.781	0.7926	0.8012	**0.8052**	**0.8096**	**0.8155**	0.8205	0.8287
ACR-SN	0.7770	0.7888	0.7953	0.8000	0.8044	0.8079	0.8144	**0.8212**	**0.8295**

Table 4. The experimental results of node classification on DBLP.

Methods	Training Ratio								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Node2vec	0.7199	0.7272	0.7295	0.7322	0.7344	0.7367	0.7347	0.7281	0.7269
LINE	0.6862	0.6894	0.6929	0.6951	0.6988	0.6999	0.7015	0.7034	0.7049
Node2vec+attr	0.7078	0.7204	0.7249	0.7298	0.7322	0.7345	0.7320	0.7256	0.7258
LINE+attr	0.6897	0.7024	0.7119	0.7136	0.7148	0.7104	0.7171	0.7118	0.7235
UPP-SNE	0.3747	0.3778	0.3793	0.3786	0.3794	0.3802	0.3798	0.3750	0.3791
Paper2vec	0.3700	0.3731	0.3772	0.3785	0.3799	0.3805	0.3791	0.3756	0.3777
GraphSAGE	0.3653	0.3680	0.3722	0.3746	0.3752	0.3780	0.3761	0.3750	0.3767
GCN	0.3957	0.3945	0.3995	0.4049	0.4172	0.4142	0.4046	0.4038	0.4056
GAT	0.5436	0.5439	0.5321	0.5374	0.5353	0.5384	0.5424	0.5449	0.5460
ACR-SN -avg	0.7292	0.7541	0.7634	0.7703	0.7732	0.7739	0.7779	**0.7829**	**0.7808**
ACR-SN	**0.7300**	**0.7553**	**0.7623**	**0.7739**	**0.7770**	**0.7748**	**0.7792**	0.7778	0.7777

Implementation Details. We implemented our method ACR-SN based on Pytorch framework. We used Adam optimizer and set the learning rate to 0.005. The epoch is set to 300 to reach a stable accuracy performance. The embedding dimension d here is set to 128, and the output layer size(number of paper areas) is 4. Similar to [9], we used a two layer GCN, the hidden layer dimension is 16. In each iteration, we used a full dataset and perform batch gradient descent. The memory usage is $\mathcal{O}(|E|)$ for the usage of sparse storage method. For the Node2vec, we set the walk length to 5, and the window size to 3. For LINE, we used both the first and second neighbors and set the negative samples to 5.

For the link prediction task, we split the 80% of total edges as train-set,10% as validation-set and the rest as test-set. For each set of edges we randomly generated the same size of negative edges that did not appear in the original graph, that is 50% true edges versus 50% false edges.

5.2 Results and Analysis

Here we utilize two tasks to validate the effectiveness of the method. 1). Node Classification: this task is to conduct the classification of papers. 2). Link Prediction: this task is to determine whether there is a citation link between two arbitrary papers. These two tasks are widely used in network embedding field. Next, we will introduce the details of these experimental results:

Node Classification. Table 3 and Table 4 illustrate the detailed results on Semantic Scholar and DBLP datasets. On Semantic Scholar dataset, the proposed ACR-SN outperforms structure-based method (Node2vec, LINE) and the combined methods (Node2vec+attr, LINE+attr, UPP-SNE, Paper2vec), which demonstrates the efficiency of our proposed method. The comparison results between these two kinds of methods demonstrate that structure feature is necessary, and attributes of nodes also play an important role in learning the node representation. The results of Node2vec+attr and LINE+attr reveal that the intuitive combination of structure feature and attributes improve the representation ability compared with structure-based methods, while pairwise attention and author attention in ACR-SN catch the features of paper more effectively. Furthermore, Our method performs better on the unique scholarly datasets compared with UPP-SNE. Also, the utilization of graph convolutional network learn the network structure information better than the CBOW model used in Paper2vec. Finally, ACR-SN achieves higher accuracy than GCN and GraphSAGE, which indicates that our model is more suitable on the scholarly datasets. But on lower training ratio, GAT gains slightly higher accuracy than ACR-SN.

As for DBLP dataset, ACR-SN gain higher accuracy than most baselines. Surprisingly, the structure-based methods (Node2vec, LINE) achieve higher accuracy than the combined methods. The addition of node attributes reduces the experimental performance, which illustrates that content and combined methods are sensitive to the scholarly datasets. The decrease in accuracy indicates that some of the baselines are also sensitive to data imbalance.

In order to demonstrate the effectiveness of attention mechanism, we compare ACR-SN with its variant ACR-SN-avg on the task of node classification, and show the experimental results on Table 3 and Table 4. ACR-SN-avg is the variant of ACR-SN without considering the second attention layer, which uses the average of author embeddings instead of the attention network. The results show that our method gains higher accuracy than the average method under the small training ratio. It demonstrates that the author's contribution attention layer can distinguish the different importance of authors, and achieve a better paper classification result.

Link Prediction. The link prediction task is to determine if there exits the citation relationship in a pair of papers based on their learned node embeddings. Figure 3 shows the link prediction results on semantic scholar dataset. As shown in Fig. 3, we observe that ACR-SN gains the highest AUC among these compared

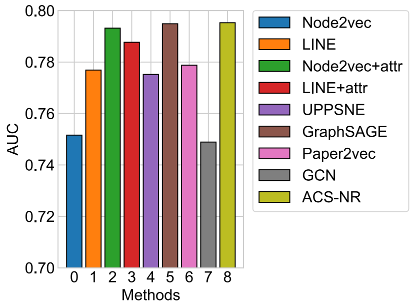

Fig. 3. Link prediction results on Semantic Scholar.

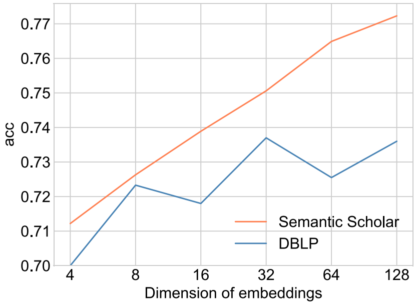

Fig. 4. Parameter Sensibility of the Embedding Dimensions.

methods. Specifically, our method ACR-SN is trained on node classification task, and we use these learned embedding on the link prediction task, which is a cross-task experiment. Comparing to the unsupervised methods like Node2vec, LINE and GraphSAGE, ACR-SN achieves the highest average precision on Semantic Scholar dataset. This result demonstrates that our method can learn the paper embedding effectively.

Table 5. The Distribution of Author Attention Weights in the paper.

Papers	Authors	Weights
Mixture Representations for Inference and Learning in Boltzmann Machines	Neil D. Lawrence	0.0932
	Christopher M. Bishop	0.8024
	Michael I. Jordan	**0.1044**
Loopy Belief Propagation for Approximate Inference: An Empirical Study	Kevin P. Murphy	0.4036
	Yair Weiss	0.2436
	Michael I. Jordan	**0.3528**

Parameter Sensibility Analysis. In our model, embedding dimension is an important parameter. So in Fig. 4, we can observe that with the embedding dimension of paper increasing, the accuracy in node classification is in a rising trend. In DBLP dataset, the accuracy shows the fluctuation. While in Semantic Scholar dataset, the performance of higher embedding dimension is better. As a result, we use the same 128 dimension of paper embedding in all experiments.

5.3 Case Study

From the scholarly dataset, we choose two papers of Michael I. Jordan, professor of UCB, to demonstrate the different contribution of authors in a paper. We select two papers [10] and [13] in his different developing phase, published in 1999 and 2013 respectively. The authors' attentions in these two papers are shown in Table 5. Michael I. Jordan plays different roles in these papers. In both papers, he is the last author. In the first paper, according to our attention calculation, he contributes about ten percent to the paper fewer than the second author, while in the second paper the attention value suggests that the authors contribute nearly averaged to this work. It suggests that the author contributes to paper in different stages differently, and the various contribution could help us to better comprehend the relationship between the authors and papers.

6 Conclusion

In this paper, we proposed a novel scholarly network embedding framework called ACR-SN, for scholarly network analysis. Specifically, we proposed two attention networks for capturing the authors' influences and contribution, respectively. Then we utilized a GCN method to model the diffusion of papers' attributes influences. Extensive experiments show the effectiveness of ACR-SN in many applications including paper classification and citation prediction.

There are still some further directions in the future. First, we would combine the citation network with co-author network. Second, we would deepen the study of co-author relationship for the scholarly network analysis.

Acknowledgements. This work was supported in part by National Natural Science Foundation of China (Grant No. 61703386), the Anhui Sun Create Electronics Company Ltd., under Grant KD1809300321, in part by the National Key R&D Program of China under Grant 2018YFC0832101, and in part by the National Key New Product Plan of China under Grant 2014GRC30006.

References

1. Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., Smola, A.J.: Distributed large-scale natural graph factorization. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 37–48. ACM (2013)
2. Ammar, W., et al.: Construction of the literature graph in semantic scholar. arXiv preprint arXiv:1805.02262 (2018)
3. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems, pp. 585–591 (2002)
4. Dong, Y., Johnson, R.A., Chawla, N.V.: Will this paper increase your h-index? Scientific impact prediction. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 149–158 (2015)
5. Du, D., et al.: Solving link-oriented tasks in signed network via an embedding approach. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 75–80. IEEE (2017)
6. Ganguly, S., Pudi, V.: Paper2vec: combining graph and text information for scientific paper representation. In: Jose, J.M., Hauff, C., Altıngovde, I.S., Song, D., Albakour, D., Watt, S., Tait, J. (eds.) ECIR 2017. LNCS, vol. 10193, pp. 383–395. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56608-5_30

7. 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. ACM (2016)
8. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems, pp. 1024–1034 (2017)
9. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
10. Lawrence, N.D., Bishop, C.M., Jordan, M.I.: Mixture representations for inference and learning in Boltzmann machines. arXiv preprint arXiv:1301.7393 (2013)
11. Lin, H., Wang, H., Du, D., Wu, H., Chang, B., Chen, E.: Patent quality valuation with deep learning models. In: Pei, J., Manolopoulos, Y., Sadiq, S., Li, J. (eds.) DASFAA 2018. LNCS, vol. 10828, pp. 474–490. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91458-9_29
12. Ma, Y., Uzzi, B.: Scientific prize network predicts who pushes the boundaries of science. Proc. Natl. Acad. Sci. **115**(50), 12608–12615 (2018)
13. Murphy, K., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: an empirical study. arXiv preprint arXiv:1301.6725 (2013)
14. Pan, S., Wu, J., Zhu, X., Zhang, C., Wang, Y.: Tri-party deep network representation. Network **11**(9), 12 (2016)
15. Pei, H., Wei, B., Chang, K.C.C., Lei, Y., Yang, B.: Geom-GCN: geometric graph convolutional networks. In: International Conference on Learning Representations (ICLR) (2020)
16. 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. ACM (2014)
17. Petersen, A.M.: Quantifying the impact of weak, strong, and super ties in scientific careers. Proc. Natl. Acad. Sci. **112**(34), E4671–E4680 (2015)
18. Sinha, A., et al.: An overview of Microsoft Academic Service (MAS) and applications. In: Proceedings of the 24th International Conference on World Wide Web, pp. 243–246. ACM (2015)
19. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1067–1077. International World Wide Web Conferences Steering Committee (2015)
20. Tao, S., Wang, X., Huang, W., Chen, W., Wang, T., Lei, K.: From citation network to study map: a novel model to reorganize academic literatures. In: Proceedings of the 26th International Conference on World Wide Web Companion, pp. 1225–1232 (2017)
21. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
22. Wang, H., et al.: A united approach to learning sparse attributed network embedding. In: 2018 IEEE International Conference on Data Mining (ICDM), pp. 557–566. IEEE (2018)
23. Wang, H., et al.: MCNE: an end-to-end framework for learning multiple conditional network representations of social network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1064–1072 (2019)
24. West, J.D., Wesley-Smith, I., Bergstrom, C.T.: A recommendation system based on hierarchical clustering of an article-level citation network. IEEE Trans. Big Data **2**(2), 113–123 (2016)

25. Xia, F., Wang, W., Bekele, T.M., Liu, H.: Big scholarly data: a survey. IEEE Trans. Big Data **3**(1), 18–35 (2017)
26. Yan, E., Ding, Y.: Scholarly networks analysis. In: Alhajj, R., Rokne, J. (eds.) Encyclopedia of Social Network Analysis and Mining, pp. 1643–1651. Springer, New York (2014). https://doi.org/10.1007/978-1-4614-6170-8_249
27. Yang, C., Liu, Z., Zhao, D., Sun, M., Chang, E.: Network representation learning with rich text information. In: Twenty-Fourth International Joint Conference on Artificial Intelligence (2015)
28. Zhang, D., Yin, J., Zhu, X., Zhang, C.: User profile preserving social network embedding. In: IJCAI International Joint Conference on Artificial Intelligence (2017)