SBiNE: Signed Bipartite Network Embedding

Youwen Zhang[1], Wei Li[1], Dengcheng Yan[1(✉)], Yiwen Zhang[1], and Qiang He[2]

[1] School of Computer Science and Engineering, Anhui University, Heifei 230601, China
wen070864@gmail.com, {91019,yanzhou,zhangyiwen}@ahu.edu.cn

[2] School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC 3122, Australia
qhe@swin.edu.au

Abstract. This work develops a representation learning method for signed bipartite networks. Recent years, embedding nodes of a given network into a low dimensional space has attracted much interest due to it can be widely applied in link prediction, clustering, and anomalous detection. Most existing network embedding methods mainly focus on homogeneous networks with only positive edges and single node type. However, negative edges are more valuable than positive edges in certain analysis tasks. Even though the work on signed network representation learning distinguishes between positive and negative edges, it does not consider the difference in node types. Moreover, bipartite network representation learning which considers two types of vertices do not tell link signs. In order to solve this problem, we further consider the link sign on the basis of the bipartite network to conduct signed bipartite network analysis. In this paper, we propose a simple deep learning framework SBiNE, short for signed bipartite network embedding, which both preserves the first-order (i.e., observed links) and second-order proximity (i.e., unobserved links but have similar sign context), and then by optimizing the objective function, experiments on three datasets show that our proposed framework SBiNE is competitive in link sign prediction task.

Keywords: Signed bipartite networks · Network embedding · Link sign prediction

1 Introduction

Not all networks in real-world are the same, and in fact, relationship between entities can be expressed via biological networks, social networks and communication networks. Most previous work have primarily focus on signed networks with single node type or bipartite networks with only positive links. However, the network structure of signed bipartite networks is often overlooked. A signed

H. Gao et al. (Eds.): CollaborateCom 2020, LNICST 349, pp. 479–492, 2021.
https://doi.org/10.1007/978-3-030-67537-0_29

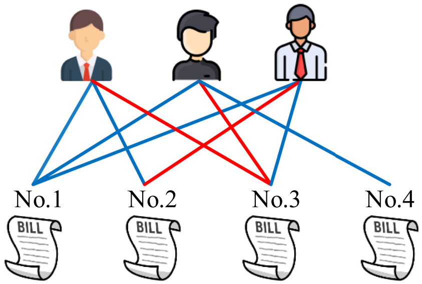

Fig. 1. Congresspersons vote "Yea" or "Nay" for the bills

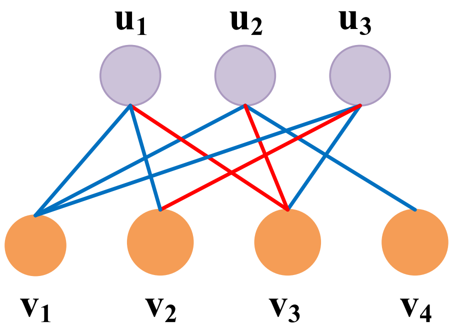

Fig. 2. The signed bipartite network corresponding to Fig. 1

bipartite network is a special network with two sets of nodes, meanwhile, edges exist only between different types of nodes. In addition, there are no edges between nodes of the same type. In fact, signed bipartite networks are involved in many domains of our lives. For example, in the United States Congress, the voting records are modeled as a signed bipartite social network (i.e., contains both positive and negative connections) between the representatives and the bills [1]. In several analytical tasks, negative links have proven to be significantly useful in improving positive link prediction [2] and recommendation performance [3].

In order to perform predictive analysis on the signed bipartite network, this requires us first to learn low-dimensional dense representations for nodes. One common form of data mining is network embedding which transforms network information into low-dimensional dense vectors, while preserving the network topology and using it as an input to existing machine learning algorithms, processing tasks such as node classification [4], link prediction [5], recommendation [6–9] and visualization [10].

Most existing signed network representation learning are based on homogeneous networks [11], which are not applicable to signed bipartite networks when having two node types. Our work investigates how to effectively learn the node embeddings which can well preserve the original network structure. Motivated by LINE [12], we introduce second-order proximity to signed bipartite networks to measure the similarity between nodes of the same type since there is no link between them. The first-order proximity is measured by the embedding of observed edges which will be learned by deep learning. For better understanding, we take Fig. 2 as an example. From the perspective of topology structure, LINE [12] and BiNE [13] will consider that user u_1, u_2 and u_3 are similar because they have multiple common neighbours. But if we take the sign between congressperson and bill into consideration, although satisfying the second-order proximity in structure, u_1 and u_3 actually are of big difference because of their different sign context. The major contributions of our method are listed as follows:

- Based on bipartite networks, we further consider the link sign difference to perform signed bipartite network representation learning.
- To measure the similarity between nodes in signed bipartite network, we design a suitable objective function and then propose a simple deep learning framework SBiNE for signed bipartite network representation learning, which learns low-dimensional vectors for nodes via optimizing the objective function.
- We conduct experiments on three real-world datasets. Experimental results demonstrate the effectiveness of the proposed framework SBiNE.

The rest of this paper is organized as follows. We introduce some related works in Sect. 2. In Sect. 3, the motivation is supplied for this article. In Sect. 4, we introduce the proposed framework SBiNE with the details about the signed bipartite network embedding objective function. In Sect. 5, we perform empirical evaluations with discussion. Finally, we present the conclusion in Sect. 6.

2 Related Work

Network representation learning or network embedding aims to learn the low-dimensional vector representation of a given network. Initially, network representation learning algorithms were mainly based on matrix feature vectors. Spectral analysis [14] algorithm obtains k-dimensional node representations by calculating the top-k eigenvectors of the Laplacian matrix. The well-known algorithm Deepwalk [15] is inspired by the word representation vector model Skip-gram in NLP. It treats the sequence of nodes as sentences and learns the node embedding from the sequence of random walks.

LINE [12] improves Deepwalk [15] and preserves first-order and second-order proximity. Moreover, it is suitable for large-scale directed, undirected, and weighted networks. GraRep [16] introduces higher-order proximity between nodes on this basis. The bipartite network embedding BiNE [13] proposed a novel optimization framework by accounting for both the explicit relations and implicit relations in learning the vertex representations. Likewise, heterogeneous information network embedding aims to embed multiple types of nodes into a low-dimensional vector space. RHINE [17] utilized the structural characteristic of Affiliation Relation(ARs) and Interaction Relation(IRs) and then proposed a novel Structure Relation-aware Heterogeneous Information Network Embedding model. However, the aforementioned methods concerning homogenous network, bipartite network, and heterogeneous network cannot directly be applied to signed networks because they don't consider negative links. Signed network embedding which considers negative links like SiNE [11] introduced structural balance theory into signed social networks and optimized the objective function based on deep learning to automatically learn signed network embedding. Other signed network representation learning, like SNEA [18], both considers the network structure and node attributes, which makes the link sign prediction task a significant improvement. Although these signed network embedding methods consider the link sign, the node type is still single so they also cannot be applied to signed bipartite network embedding analysis. Thus, we focus on the problem of learning low-dimensional vectors for nodes in the signed bipartite networks by utilizing the power of deep learning. More specifically, we design a simple framework SBiNE which optimizes a multi-layer perceptron based objective function to learn signed bipartite network embedding automatically.

3 Motivation

In this section, we briefly illustrate the motivation of our research according to Fig. 1 and Fig. 2. Specifically, Sect. 3.1 discusses the necessity of studying signed bipartite networks, and Sect. 3.2 explains why first-order and second-order proximity should be introduced into signed bipartite networks.

3.1 Why Should Signed Bipartite Networks Be Researched?

The relationship between entities can be represented by networks, and different networks have different characteristics. For example, for a homogeneous network of social relationships between blog authors on the BlogCatalog website, node2vec [19] designed a biased random walk to effectively explore various node neighbors. For a bipartite network containing authors and publishers in DBLP. BiNE [13] learns the representation vector of nodes from two perspectives of explicit and implicit relations. The two-layer relationship between users and items in e-commerce viewing and purchasing constitutes a multi-dimensional bipartite network, hence MINES [20] aims to learn the representation of nodes in each dimension of the network structure. Heterogeneous information network of entities, words and categories in Wikipedia, Zhao et al. [21] build the co-occurrence matrices between same and different types of nodes, and use coordinate matrix factorization to jointly learn the representations of entities, words and categories from all matrices. The links of trust and distrust between people in Epinions form a signed social network, and SiNE [11] uses triangular structure balance theory to design the objective function and optimize it.

The previous network representation learning work has ignored a common form of the signed network—signed bipartite network, which inherits the advantages of signed networks with negative edges that indicate distrust or dislike, and there are two sets of nodes. It should be noted that links only exist between nodes of different types.

3.2 Why Should the First-Order and Second-Order Proximity Be Introduced?

We take Fig. 2 as an example to clearly illustrate these two proximities in signed bipartite networks. Purple nodes are the representatives in the United States Congress and orange nodes are the bills they have voted on. Blue positive link or red negative link denotes a congressperson voted "Yea" or "Nay" for the bill respectively. Naturally, the observed links in Fig. 2 preserve the first-order explicit relationship between the node sets of U and V. u_1 and v_1 node pair is such the first-order proximity relationship. Apparently, those observed links are not sufficient for preserving the global network structures. However, the second-order proximity of the signed bipartite network is quite different from the previous unsigned homogenous network. Node v_1, v_2 and v_3 are the common neighbors of u_1 and u_3. From the perspective of LINE [12], it assumes that u_1 and u_3 have several neighbor nodes in common, so they are supposed to be similar. However, if we consider the link sign context, the result will be different. Node v_1 is the common bill that u_1 and u_3 vote, but these two persons show different voting preferences on v_2 and v_3. Hence, although u_1 and u_3 have similar neighbors, we assume that they are still dissimilar. We apply the proven equation [22] to fit the problem of second-order implicit relationship between the node sets of U or V.

In study [22], it proved that LINE(2nd) is actually factoring two different matrices separately. For each directed edge (v_i, v_j), it defines v_j as the "context" of v_i. V denote a matrix whose i-th column is the vertex embedding $\mathbf{v}_i$ and U denote a matrix whose j-th column is the "context" embedding $\mathbf{u}_j$. It figured out that LINE(2nd) is factoring a matrix $M^{(2)} = V^T U$. According to the objective function in LINE(2nd), it characterizes the matrix $M^{(2)}$ that LINE(2nd) is actually factoring:

$$M_{ij}^{(2)} = PMI(v_i, v_j) - logk \tag{1}$$

4 Signed Bipartite Network Embedding

We describe the framework SBiNE to preserve the explicit relationship with the positive or negative sign between two sets of nodes and the implicit relationship to measure the same set of nodes with similar sign context since there is no link between them. Experiments show that neither of these two conditions can be missed. Considering the first-order proximity relationship between representatives and bills is far more not enough, the similarity between congresspersons also does important. Node vectors are used to denote the features of congresspersons and bills, and edge embeddings represent the "Yea" or "Nay" relationship between a congressperson and a bill. First, we summarize the major notations used throughout this paper. Next, the objective function designed for signed bipartite networks will be introduced and then we will explain the details of the simple framework SBiNE.

4.1 Problem Formulation

Notations. Given a signed bipartite network G = (U, V, E) where U = $\{u_1,u_2,\ldots u_m\}$ and V = $\{v_1,v_2,\ldots v_n\}$ is the set of the m, n nodes in G respectively. E denotes the set of observed edges between the nodes in the set U and V. Each edge $e_{ij} \in E$ is associated with a sign $\mathbf{A}_{ij}$. $\mathbf{A}$ is the $m \times n$ adjacency matrix of G. We use $\mathbf{A}_{ij} = 1$ or -1 to represent a positive or negative link between u_i and v_j , and $\mathbf{A}_{ij}$=0 denotes no link.

Problem Definition. The purpose of designing the framework SBiNE is to effectively learn the representation of the nodes in the signed bipartite network, so as to use the learned vectors of two sets of nodes (i.e., $\mathbf{X}^{M\times d}$ and $\mathbf{Y}^{N\times d}$) as input to the link sign prediction task. Each row of $\mathbf{X}$ is an embedding vector with d features and the same goes for $\mathbf{Y}$. In the embedding Euclidean space, the observed explicit links in the signed bipartite graph and the implicit relationships between the set of U or V should be properly preserved. The input and output of SBiNE can be defined as follows:

Input: A signed bipartite network G = (U, V, E) and its adjacency matrix $\mathbf{A}$.

Output: D-dimensional embedding vector for each node u_i ,v_j.

4.2 First-Order and Second-Order Proximity in Signed Bipartite Network

Some recent studies on signed networks have shown that negative and positive edges have significantly different properties. Therefore, we cannot directly apply the method of unsigned network representation learning to the signed network. In addition, there are two set of nodes in the signed bipartite network. Therefore, we design a novel applicable objective function for the special network of signed bipartite networks. We first clarify the proximity that exists in the signed bipartite network, which is also our motivation for constructing the objective function.

Condition 1: second-order proximity between nodes of the same type. It measures the similarity between u_i and u_k, and is used to distinguish nodes of the same type that have common neighbors but quite different sign patterns.

Condition 2: first-order proximity according to the observed links composed of node sets of U and V. To measure the relation between u_i and v_j, we learn the edge vector $\mathbf{E}_{ij}$ for them.

Previous work [18] show that the second-order proximity actually factorizes the Pointwise Mutual Information (PMI) [23] matrix of each node pair with a constant shift. MSE is chosen as the loss function, so the objective function of the second-order implicit relationship can be written as:

$$O_1 : \arg\min \sum_{e_{ij} \in E} \frac{1}{|E|} \left(f\left(\mathbf{x}_i\right) \cdot f\left(\mathbf{y}_j\right) - \mathrm{PMI}(i,j) + \log k \right)^2 \tag{2}$$

Since the signed bipartite network is an undirected network, each edge can be treated as two directed edges with opposite directions. Therefore, we convert the expression of PMI into:

$$PMI(i,j) = \log \frac{\sum_{p=1}^{m} \deg\left(u_p\right) + \sum_{q=1}^{n} \deg\left(v_q\right)}{\deg\left(u_i\right) \deg\left(v_j\right)} \tag{3}$$

Where $\deg\left(u_i\right)$ and $\deg\left(v_j\right)$ are the degrees of nodes u_i and v_j, respectively. In [22], to avoid time-consuming, k defines the negative samples. We empirically set $k = 15$ in this work. The $\mathbf{x}_i$ and $\mathbf{y}_j$ in expression (1) are the d-dimensional embeddings of nodes u_i and v_j, respectively, and will be continuously learned during the training process of the framework SBiNE. The mapping function f performs dimensionality reduction operations on $\mathbf{x}_i$ and $\mathbf{y}_j$ respectively, and after obtaining new feature vectors, the dot product is performed between the node pairs. Details about the function f will be discussed in the following subsection.

For condition 2, given that there are two kind of edges in the signed bipartite network: positive edges and negative edges, we also assume that the sign of the edges follows the Bernoulli distribution as in work [24], either -1, otherwise 1. The feature of the edge is the relationship between the learning nodes u_i and v_j. The edge representation vector $\mathbf{E}_{ij}$ adopts the element-wise product of two node embedding $\mathbf{x}_i$ and $\mathbf{y}_j$ after the dimensionality reduction of the mapping function f as input. The objective function of condition 2 is shown in Eq. (4).

$$O_2 : \arg\min \sum_{e_{ij} \in E} \frac{1}{|E|} \log g\left(\mathbf{E}_{ij}\right)^{\frac{1+\mathbf{A}_{ij}}{2}} \left(1 - g\left(\mathbf{E}_{ij}\right)\right)^{\frac{1-\mathbf{A}_{ij}}{2}} \tag{4}$$

Considering the learned node embeddings are supposed to satisfy above two proximities, we can finally obtain the objective function as follows:

$$\begin{aligned} O : \arg\min \sum_{e_{ij} \in E} \frac{\beta}{|E|} \left(f\left(\mathbf{x}_i\right) \cdot f\left(\mathbf{y}_j\right) - PMI(i,j) + \log k \right)^2 \\ - (1-\beta) \log g\left(\mathbf{E}_{ij}\right)^{\frac{1+\mathbf{A}_{ij}}{2}} \left(1 - g\left(\mathbf{E}_{ij}\right)\right)^{\frac{1-\mathbf{A}_{ij}}{2}} \end{aligned} \tag{5}$$

The mapping function g models the relationship between the edge representation vector space and the link sign. Details will be explained in the framework SBiNE. The hyper-parameter $\beta \in [0,1]$ is used to control the influence of these two conditions during the process of learning node representations (Fig. 3).

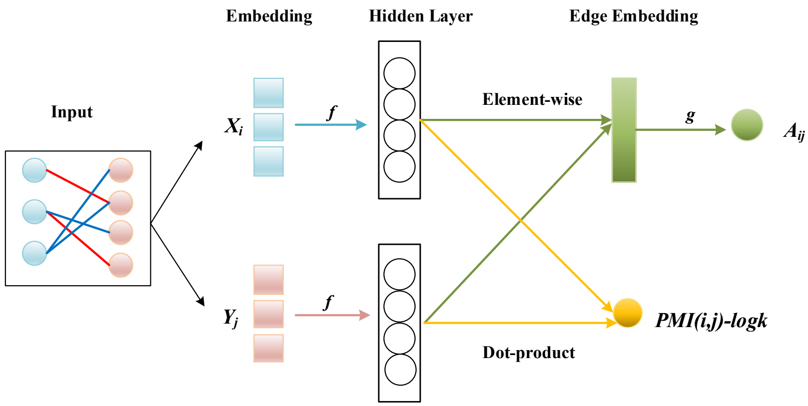

Fig. 3. The architecture of the proposed deep learning framework SBiNE

4.3 The Architecture of SBiNE

Since we have the objective function, our following task is to learn well-represented vectors for the nodes in the signed bipartite network, as well as suitable nonlinear mapping functions f and g. Deep learning technology provides strong technical support for nonlinear representation learning [25]. We use multi-layer perceptron to model the nonlinear mapping functions f and g. By optimizing the objective function (5), the node embeddings and the mapping functions can be learned simultaneously via back-propagation, since the node embedding is integrated into the neurals.

The input to the framework is a signed bipartite network G with two sets of initial node embeddings $\mathbf{X}_i$ and $\mathbf{Y}_j$. For better understanding, we first give an explanation of the hidden layer. The outputs of the hidden layer of the two neural networks are given as:

$$\begin{aligned} f\left(X_i\right) &= \tanh\left(W^{11}X_i + b^{11}\right) \\ f\left(Y_i\right) &= \tanh\left(W^{12}Y_i + b^{12}\right) \end{aligned} \tag{6}$$

Where tanh is the hyperbolic tangent function, other activation functions like sigmoid, ReLU are also effective. $\mathbf{W}^{11}$ and $\mathbf{W}^{12}$ are the weights of the hidden layer and b^{11}, b^{12} are the bias. The primary function of the hidden layer is to reduce the dimension of the node embeddings on the basis of retaining effective features. The first step is to achieve the second-order proximity in the signed bipartite network. With regard to the node pair $(u_i\ ,v_j)$, after a dot product is operated on $f(\mathbf{x}_i)$ and $f(\mathbf{y}_j)$. We hope its value as close as possible to $PMI(i,j)-logk$ through back propagation. The loss function here uses the mean square error MSE. The second step is to achieve the first-order proximity between the node pair (u_i,v_j) in signed bipartite network. We use edge embedding to represent this first-order proximity relationship which will be well learned via deep learning. We perform element-wise operation on $f(\mathbf{x}_i)$ and $f(\mathbf{y}_j)$, and then get their edge embedding $\mathbf{E}_{ij}$. $\mathbf{E}_{ij}$ will be the input of the new single neural with sigmoid nonlinearity. The output of the neural is:

$$g(\mathbf{x}_i, \mathbf{y}_i) = \text{sigmoid}(\mathbf{E}_{ij}) = \text{sigmoid}\left(f(\mathbf{x}_i) \odot f\left(\mathbf{y}_j\right)\right) \tag{7}$$

5 Experimental Results

We empirically evaluate our proposed signed bipartite network embedding framework SBiNE. By applying the method to three real-world datasets, we seek to answer the following two questions. First, does network representation learning for signed bipartite network provide an improvemance for predicting link signs? Second, how does dimension d of the learned vectors affect the performance of the model? For better understanding, we also conduct parameter sensitivity analysis.

5.1 Datasets

The three datasets used in the experiment are all from [18]. Bonanza is similar to e-commerce sites like Amazon or eBay, where every user can buy and sell a variety of goods. A buyer can rate the product purchased from a seller when a transaction is finished. To distinguish between the buyer and the seller, we use the node sets U and V to represent them. U.S.Senate and U.S.House datasets represent the role call votes combined from 1st to 10th United States Congress. We represent the senators or representatives by the set U and the bills that were voted by V. The positive link indicates that the senate support a bill, the negative link is the opposite. Table 1 has detailed information about these three datasets. Apparently, it can be seen that Bonanza is an extremely unbalanced dataset with only 2.02% negative links.

Table 1. Statistics of the three datasets

Dataset	m	n	Positive links	Negative links	$\lvert E\rvert$
Bonanza	7,919	1,973	97.98%	2.02%	36.543
U.S.Senate	1,056	145	98.88%	44.69%	27.083
U.S.House	1,281	515	53.96%	46.04%	114.378

5.2 Experimental Settings

Next, we first discuss the settings used for above three datasets on task of link sign prediction. For all three datasets, 80% of all the links are randomly selected as training set , so the remaining 20% used to test the model performance. To the best of our knowledge, only [26] is devoted to proposing balance theory in signed bipartite networks to predict link signs. Therefore, this work is the first study to carry out representation learning research on the signed bipartite networks. In order to compare with [26], we also use both F1 and AUC for evaluation.

5.3 Comparison Results

We seek to find out whether the signed bipartite network representation learning can improve the mining task of predicting the link sign on the three datasets. The task of link sign prediction is to predict the sign of the given links. In Tables 2 and 3, we can see the comparison results of AUC and F1. Because of the high imbalance of the Bonanza dataset, we will discuss the results of link sign prediction of this dataset separately. The definitions of the related six baseline methods [26] are as follows:

- **Degree Based Supervised Classifier (SCd):** By extracting features from each node, Scd trains the model by constructing the training dataset composed of links of positive and negative semantic links. With the trained model, it can predict arbitrary link sign.
- **Signed Catterpillars Based Supervised Classifier (SCsc):** Compared with Scd, this method extract features from characteristics of neighbor nodes based on balance theory.
- **Matrix Factorization (MF):** A traditional matrix factorization method to predict the unknown link sign by optimizing the node feature vectors via biadjacency matrix.
- **Matrix Factorization with Balance Theory (MFwBT):** It calculates if using balance theory could suggest a positive or negative link for nodes that do not exist in the network.
- **Lazy Random Walk (LazyRW):** LazyRW which is used to as a comparison against the SBRW performs a random walk to the neighbor node of the current node in a probabilistic way.
- **Signed Bipartite Random Walk (SBRW):** This method performs random walk on the unipartite signed network constructed by an adjacency matrix.

In the latter two datasets, our SBiNE model achieved the best performance on both AUC and F1 metrics for predicting the link sign. To be specifically, AUC and F1 are higher than the highest values of AUC and F1 of the other six methods, respectively.

Table 2. Results of predicting the link sign in terms of AUC and F1 metrics

Metric	Algorithm	Bonanza	U.S.Senate	U.S.House
AUC, F1	SCd	(0.553,0.959)	(0.638,0.654)	(0.625,0.635)
	SCsc	(**0.664**,0.674)	(0.812,0.823)	(0.827,0.837)
	MF	(0.593,0.903)	(0.792,0.812)	(0.831,0.846)
	MFwBT	(0.608,0.905)	(0.814,0.827)	(0.834,0.848)
	LazyRW	(0.547,**0.979**)	(0.808,0.821)	(0.815,0.827)
	SBRW	(0.582,0.949)	(0.836,0.849)	(0.846,0.858)
	SBiNE	(0.626,0.954)	**(0.915,0.857)**	**(0.934,0.869)**

In the extremely unbalanced dataset of Bonanza, we observed that among all the six comparison methods except SCsc and MFwBT, F1 are all greater than 0.9, but AUC is less than 0.6. Obviously, this classification results have bad performance. To further analyze the problem of highly imbalance in the Bonanza dataset, we made three sets of experiments and compared them with the six baseline methods. By adding different weights to the positive and negative sample categories in the loss function, we obtain three sets of results. When AUC and F1 are relatively balanced, AUC in SBiNE is second only to SCsc, and F1 is second only to SCd and LazyRW. However, both AUC and F1 performs better than MFwBT. If F1 is higher, the pair of (AUC, F1) is also better than LazyRW with the highest F1 in the six methods. According to the third set of results, we try to increase the AUC value by training as much as possible, then the results obtained are also higher than the SCsc with the highest AUC in the six methods.

Table 3. Results of link sign prediction with regard to Bonanza dataset

Algorithm	SCd	SCsc	MF	MFwBT	LazyRW	SBRW	SBiNE		
AUC	0.553	0.664	0.593	0.608	0.547	0.582	0.626	0.555	**0.668**
F1	0.959	0.674	0.903	0.905	0.979	0.949	0.954	**0.981**	0.886

5.4 Impact of Dimension d

In order to analyze the impact of dimension d on SBiNE, we set k = 15 as a fixed value, and $\beta = 0.05$. Meanwhile, we vary d as $\{8, 16, 32, 64, 100, 128, 200\}$. Bonanza is an extremely unbalanced dataset with only 2.02% of negative links, so we conducted this experiment on the two other datasets. From (a) (b), we observe that if d is too small, for example, when the dimension is 8, the F1 of the two datasets is very low, then the learned node feature vectors cannot adequately represent the structural characteristics of the original node. When $d \in [8, 32]$, the value of F1 and AUC of both datasets are improving. The U.S. Senate dataset slowly decreases after the AUC reaches its peak when the dimension is 32, while F1 slowly increases and then decreases. The AUC and F1 of the U.S. House dataset rapidly decreased and then increased after reaching the peak, and then slowly declined. When the dimension d is too large, it tends to overfit. Considering the moderate size of the dataset, we finally set the dimension of the vector to a reasonable value of 32.

5.5 Parameter Sensitivity Analysis

Since k has little effect on the results in the link sign prediction, we empirically set k to 15. So in this section, we performed hyper-parameter analysis on all three datasets to observe the effect of β on the model performance. The hyper-parameter $\beta \in [0, 1]$ is mainly affected by two conditions that control the proximity of nodes in the signed bipartite network. When $\beta = 1$, SBiNE only uses the first condition, that is, the second-order proximity of the nodes in the bipartite network; when $\beta = 0$, SBiNE only uses the second condition; when $\beta \in (0,1)$, the method combines two conditions at the same time. The influence of β on the link sign prediction task is shown in Table 4 above. We can see from the Table 4 that when $\beta < 1$, F1 and AUC are better than those when $\beta = 1$. It shows that we cannot just consider the second-order proximity between nodes of the same type. When $\beta = 0.05$, SBiNE considers two conditions at the same time. Meanwhile, the classification effect of link sign prediction performs best. It indicates that second-proximity is important but not as important as first-proximity (Fig. 4).

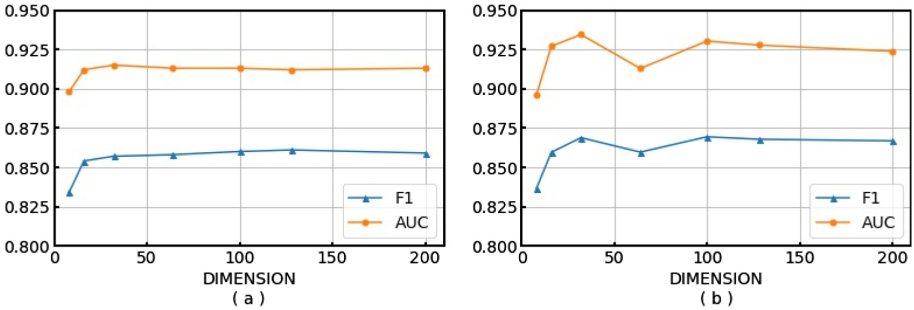

Fig. 4. The impact of embedding dimension d on SBiNE for link sign prediction

Table 4. The sensitivity of SBiNE to β in link sign prediction

Dataset	Metric	β				
		0	0.05	0.1	0.5	1
Bonanza	AUC	0.948	**0.954**	0.716	0.624	0.630
	F1	0.595	0.626	0.660	0.693	**0.755**
U.S.Senate	AUC	0.913	**0.915**	0.912	0.911	0.510
	F1	0.855	**0.857**	0.854	0.854	0.711
U.S.House	AUC	0.933	**0.934**	0.933	0.856	0.507
	F1	0.866	**0.869**	0.867	0.797	0.691

6 Conclusion

In this paper, we are committed to applying network representation learning to a special network called signed bipartite network. Furthermore, we introduce a new object function suitable for signed bipartite network embedding and propose a simple framework SBiNE to optimize it. According to the analysis of link sign prediction on the above three signed bipartite networks, we get the following two conclusions. First, the learned low-dimensional node vectors can effectively preserve the original structural topology. Through the construction of our framework SBiNE, the Hadamard product of the learned node embeddings can significantly distinguish the positive and negative links compared to [26]. Structural balance theory [27] in signed social networks has been extensively applied to improve the performance across measuring [28] and mining applications [29]. However, we can't find the trace of the triangle in signed bipartite networks due to the existence of heterogeneous nodes. In our future work, we decide to adopt balance in signed bipartite networks [26] for signed bipartite network embedding analysis.

Acknowledgement. This work was supported by the National Key Research and Development Program of China (No. 2019YFB1704101), the National Natural Science Foundation of China (no. 61872002U1936220) and the Natural Science Foundation of Anhui Province of China (no. 1808085MF197).

References

1. Derr, T., Tang, J.: Congressional vote analysis using signed networks. In: 2018 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 1501–1502. IEEE (2018)
2. Leskovec, J., Huttenlocher, D., Kleinberg, J.: Predicting positive and negative links in online social networks. In: Proceedings of the 19th International Conference on World Wide Web, pp. 641–650 (2010)
3. Ma, H., Lyu, M.R., King, I.: Learning to recommend with trust and distrust relationships. In: Proceedings of the Third ACM Conference on Recommender Systems, pp. 189–196 (2009)
4. Bhagat, S., Cormode, G., Muthukrishnan, S.: Node classification in social networks. In: Aggarwal, C. (ed.) Social Network Data Analytics, pp. 115–148. Springer, Boston (2011). https://doi.org/10.1007/978-1-4419-8462-3_5
5. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. J. Am. Soc. Inform. Sci. Technol. **58**(7), 1019–1031 (2007)
6. Zhang, Y., et al.: Covering-based web service quality prediction via neighborhood-aware matrix factorization. IEEE Trans. Serv. Comput. (2019)
7. Zhang, Y., Cui, G., Deng, S., Chen, F., Wang, Y., He, Q.: Efficient query of quality correlation for service composition. IEEE Trans. Serv. Comput. (2018)
8. Zhang, Y., Yin, C., Wu, Q., He, Q., Zhu, H.: Location-aware deep collaborative filtering for service recommendation. IEEE Trans. Syst. Man Cybern.: Syst. (2019)
9. Zhang, Y., Zhou, Y., Wang, F., Sun, Z., He, Q.: Service recommendation based on quotient space granularity analysis and covering algorithm on spark. Knowl.-Based Syst. **147**, 25–35 (2018)
10. Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
11. Wang, S., Tang, J., Aggarwal, C., Chang, Y., Liu, H.: Signed network embedding in social media. In: Proceedings of the 2017 SIAM International Conference on Data Mining, pp. 327–335. SIAM (2017)
12. 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 (2015)

13. Gao, M., Chen, L., He, X., Zhou, A.: Bine: Bipartite network embedding. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 715–724 (2018)
14. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems, pp. 585–591 (2002)
15. 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)
16. Cao, S., Lu, W., Xu, Q.: GraRep: learning graph representations with global structural information. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 891–900 (2015)
17. Lu, Y., Shi, C., Hu, L., Liu, Z.: Relation structure-aware heterogeneous information network embedding. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 4456–4463 (2019)
18. Wang, S., Aggarwal, C., Tang, J., Liu, H.: Attributed signed network embedding. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 137–146 (2017)
19. 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)
20. Ma, Y., Ren, Z., Jiang, Z., Tang, J., Yin, D.: Multi-dimensional network embedding with hierarchical structure. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 387–395 (2018)
21. Zhao, Y., Liu, Z., Sun, M.: Representation learning for measuring entity relatedness with rich information. In: Twenty-Fourth International Joint Conference on Artificial Intelligence (2015)
22. Wang, Q., Wang, Z., Ye, X.: Equivalence between line and matrix factorization. arXiv preprint arXiv:1707.05926 (2017)
23. Church, K.W., Hanks, P.: Word association norms, mutual information, and lexicography. Comput. Linguist. **16**(1), 22–29 (1990)
24. Song, W., Wang, S., Yang, B., Lu, Y., Zhao, X., Liu, X.: Learning node and edge embeddings for signed networks. Neurocomputing **319**, 42–54 (2018)
25. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
26. Derr, T., Johnson, C., Chang, Y., Tang, J.: Balance in signed bipartite networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1221–1230 (2019)
27. Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: Sitting closer to friends than enemies, revisited. In: Rovan, B., Sassone, V., Widmayer, P. (eds.) MFCS 2012. LNCS, vol. 7464, pp. 296–307. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32589-2_28
28. Derr, T., Aggarwal, C., Tang, J.: Signed network modeling based on structural balance theory. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 557–566 (2018)
29. Chiang, K.Y., Natarajan, N., Tewari, A., Dhillon, I.S.: Exploiting longer cycles for link prediction in signed networks. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 1157–1162 (2011)