"Bridge" Enhanced Signed Directed Network Embedding

Yiqi Chen
Wuhan University
Wuhan, Hubei, China
yiqic16@whu.edu.cn

Tieyun Qian*
Wuhan University
Wuhan, Hubei, China
qty@whu.edu.cn

Huan Liu
Arizona State University
Tempe, Arizona, USA
huan.liu@asu.edu

Ke Sun
Wuhan University
Wuhan, Hubei, China
sunke1995@whu.edu.cn

ABSTRACT

Signed directed networks with positive or negative links convey rich information such as like or dislike, trust or distrust. Existing work of sign prediction mainly focuses on triangles (triadic nodes) motivated by balance theory to predict positive and negative links. However, real-world signed directed networks can contain a good number of "bridge" edges which, by definition, are not included in any triangles. Such edges are ignored in previous work, but may play an important role in signed directed network analysis.

In this paper, we investigate the problem of learning representations for signed directed networks. We present a novel deep learning approach to incorporating two social-psychologic theories, balance and status theories, to model both triangles and "bridge" edges in a complementary manner. The proposed framework learns effective embeddings for nodes and edges which can be applied to diverse tasks such as sign prediction and node ranking. Experimental results on three real-world datasets of signed directed social networks verify the essential role of "bridge" edges in signed directed network analysis by achieving the state-of-the-art performance.

CCS CONCEPTS

• **Information systems → Data mining**;

KEYWORDS

signed directed network embedding, balance theory, status theory

ACM Reference Format:

Yiqi Chen, Tieyun Qian, Huan Liu, and Ke Sun. 2018. "Bridge" Enhanced Signed Directed Network Embedding. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3269206.3271738

1 INTRODUCTION

Recent years have witnessed the proliferation of online signed directed networks. For example, the consumer review sites like Epinions allow members decide whether to trust each other; the e-commerce websites such as Amazon let members express their likes and dislikes toward the purchased products. In these networks, the relations between entities convey rich information and are signed positively or negatively. The positive signs may show trust and agreement while negative ones may represent distrust and disagreement.

It may be essential to identify the positive or negative relationship between two users. For example, when recommending friends for a user u in a social network, it is a good idea not to list u's foes as candidates. The task of sign prediction aims to infer the unobserved attitudes among users, and can be formalized as predicting the sign of an edge in the network.

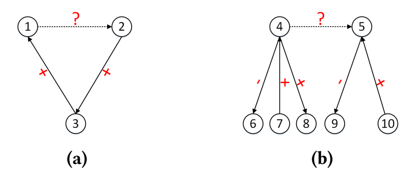


Figure 1: A triangle and a "bridge" edge

Table 1: Fractions of "bridge" edges in three online networks

Dataset	Slashdot	Epinions	Wikirfa
Fraction	47.90%	20.43%	6.45%

Sign prediction underlies many applications like recommendation, advertisement, and community detection. Most efforts [7, 10, 13, 19, 23] are devoted to exploring properties of triangles (triadic nodes) following balance theory [6] to predict positive and negative links. Figure 1a shows an example of sign prediction in a triangle. Given that u_2 likes u_3 and u_3 likes u_1, what's the sign of $\overrightarrow{e_{12}}$? The triangles are effective in capturing relationships among three users. However, in reality, there often exist a good number of "bridge" edges which are not included in any triangles like $\overrightarrow{e_{45}}$ in Figure 1b.

Assuming we have a signed directed graph $G(V, E)$ with no self-loops, where V and E denotes the node and edge set, respectively, the "bridge" edge set E_{bri} consists of the edges whose adjacent nodes do not share any common neighbors.

$$E_{bri} = \{\overrightarrow{e_{ij}} | N(v_i) \cap N(v_j) = \phi\}, \tag{1}$$

where $N(v_i)$ is the set of neighbor nodes, i.e., all nodes linking to or being linked by v_i. Meanwhile, the "triangle" edge set E_{tri} consists of the edges whose adjacent nodes share at least one common neighbor.

$$E_{tri} = \{\overrightarrow{e_{ij}} | N(v_i) \cap N(v_j) \neq \phi\}, \tag{2}$$

By definition, the "bridge" edge set E_{bri} is the complement of the "triangle" edge set E_{tri}, i.e., $E_{bri} \bigcup E_{tri} = E$. While most of edges in the signed directed network can be included in triangles, the "bridge" edges are globally present as well. Table 1 shows such evidence of "bridge" edges in three online signed directed networks.

*Corresponding author

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10...$15.00
https://doi.org/10.1145/3269206.3271738

The nodes in a triangle are connected with each other and the sign prediction for edges in a triangle can be effectively conducted based on balance theory. For example, the sign of edge $\overrightarrow{e_{12}}$ in Fig. 1a can be inferred based on the principle of "a friend of my friend is my friend" in balance theory. In contrast, the "bridge" edges lack the triangle information, and cannot be modeled using balance theory. Hence the sign prediction for "bridge" edges is a challenging task. To the best of our knowledge, none of existing methods considers "bridge" edges in signed network analysis.

In this paper, we study the problem of signed directed network embedding. We are particularly interested in how to model "bridge" edges in addition to triangles modeled by balance theory [6] and how they can be combined. To this end, we resort to status theory [5, 14] for directional edges to model "bridge" edges. We further design a deep neural network to incorporate "bridge" and triangle edges in a complementary manner. Our framework simultaneously learns embeddings for nodes and edges. With these embeddings, various social computing tasks such as sign prediction and node ranking can be carried out effectively. We conduct extensive experiments on three online networks. Results demonstrate that modeling bridge edges can help sign prediction and node ranking tasks by the performance improvements over the state-of-the-art baselines. The main contributions of this paper are as follows.

- We propose a novel signed directed network embedding model which incorporates both balance and status theories in a complementary manner.
- Our model leverages "bridge" edges in the absence of triangle information to learn effective representations for them.
- The embeddings for nodes and edges are applicable to diverse tasks such as sign prediction and node ranking.

The rest of the paper is structured as follows. In Section 2, we present the related work. In Section 3, we introduce our BESIDE model. In Section 4, we show the experimental evaluation on sign prediction. In Section 5, we conduct experimental evaluation on node ranking. We conclude the paper in Section 6.

2 RELATED WORK

We discuss the related work in two areas: network embedding approaches, and methods for sign prediction and node ranking tasks.

Network Embedding Approaches: Network embedding aims at learning a low-dimensional dense vector for each node in the network. The representation can be applied to many different tasks of network analysis like node classification [1], link prediction [15] and community detection [16]. A number of methods have been developed in this area, including DeepWalk [17], LINE [20], Node2Vec [4], and GCN [11]. Due to the power of deep neural network, these approaches show significantly better performance than the traditional methods. All above approaches are developed for unsigned network. SNE [27], SiNE [23], SIDE [10], and Sign2Vec [7] learn embeddings for signed network. These methods show improvements on signed network analysis, but they all have some drawbacks. SNE utilizes log-bilinear with random walk sampling to generate embeddings, but it does not exploit any social theory of signed networks. SiNE is guided by balance theory and aims to reflect the relationships among users, their friends and foes, but it is devised for undirected signed networks. SIDE and Sign2Vec both combine balance theory with specialized random walk sampling techniques in directed signed networks. However, they neglect the special structural property, i.e., the existing of "bridge" edges, in real networks. In contrast, our framework takes both the "bridge" edges and triangles into consideration and connects them with balance and/or status theory.

Social Computing Tasks - Sign Prediction and Node Ranking: Sign prediction and node ranking are two of the most important social computing tasks in signed network analysis. For the *sign prediction* task, all signed network embedding methods SNE [27], SiNE [23], SIDE [10] and Sign2Vec [7] carry out this experiment. In addition, FExtra [13] is a feature-engineering method focused on the structure of signed network, it chooses 23 features for each edge in the triangles representing balance and status theories. Several other approaches are developed for signed network with extra information [22, 26]. The problem investigated in our paper has the same setting with that in FExtra [13], SiNE [23], SIDE [10] and Sign2Vec [7] which does not require additional information.

For the *node ranking* task, there are mainly two types of methods. One modifies traditional ranking methods to signed network. For example, Exp [21], PageTrust [9], Modified PageRank (MPR) [18] and Modified HITS (MHITS) [25] are based on original PageRank [3] or HITS [12] and take sign links into consideration. The other is to propose a new model. For example, Prestige [28] combines positive and negative incoming links together to give each node a prestige value. Troll-Trust [25] uses a Bernoulli distribution to characterize each user as either being trustworthy or being a troll. SRWR [8] introduces the sign information into the random walk process to get personalized trust or distrust rankings.

The aforementioned methods are either developed for sign prediction or node ranking task. None of them can be directly used to solve the two problems at the same time. Our proposed approach can deal with the sign prediction and node ranking tasks simultaneously since it learns both the node and edge representations.

3 THE PROPOSED BESIDE MODEL

In this section, we first introduce the balance and status theories. Then, we present our BESIDE ("Bridge" Enhanced Signed Directed Network Embedding) model based on these two theories.

3.1 Balance Theory and Status Theory

Balance and status are two fundamental theories in social sciences. **Balance theory** is originally defined for undirected networks to model the relations of likes and dislikes [6]. It implies that "the friend of my friend is my friend" and "the enemy of my enemy is my friend". **Status theory** [5, 14] is proposed to represent the social status of the people in directed networks, where the status may denote the relative prestige, ranking, or skill level. For example, a positive/negative link from a to b denotes "b has higher/lower status than a".

Balance and status theory for "triangle" edges. Balance theory involves three users in the network and is usually modeled with triads [14, 19]. Status theory normally reflects relations between two users. However, the status relation should be transitive, which means that "a person a respected by b should be respected by b's subordinate c". This indicates that the implicit relationship between a and c can be derived via their common neighbor b using status theory. Note that here users a, b, and c also form a triad. Hence we adopt the triangle representations to infer the sign of the third edge e_{ij} in a triad using balance and status theory. Figure 2 shows all the possible types of triads.

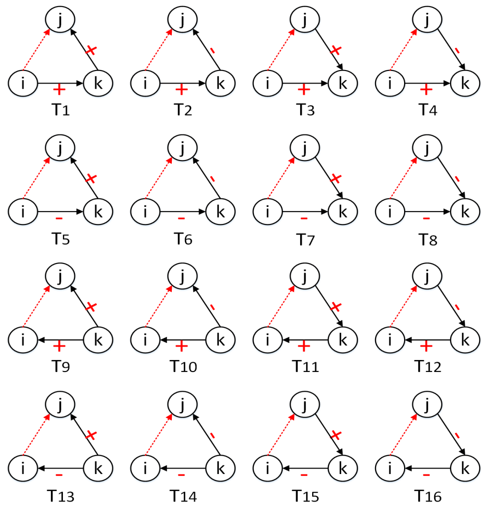


Figure 2: 16 types of signed triangles.

We now take T_1 in Figure 2 as an example. Balance theory indicates that the sign of edge e_{ij} in T_1 should be a "+" (v_j is v_i's friend) given that v_k is a friend of v_i and v_j is a friend of v_k. Similarly, status theory suggests that the sign of edge e_{ij} should be a "+" (the status of v_j is higher than that of v_i) given that v_j has higher status than v_k and v_k has higher status than v_i.

While in the above triad T_1 balance theory agrees with status theory in predicting the sign of e_{ij}, in some cases balance and status theory may contradict. For example, if T_6 in Figure 2 satisfies the status theory, i.e., the sign of e_{ij} is a "-", T_6 will not satisfy the balance theory because it breaks the principle "the enemy v_j of my enemy v_k is my friend" (from v_i's point of view). Table 2 summarizes the balance and status theory observed in the triangles and "+" or "-" denotes the satisfied prediction according to the corresponding theory. We observe that status theory may provide uncertain answers (shown as "+/-"). For example, for T_3 in Figure 2, status theory indicates that the status of v_i and v_j are both lower than v_k, thus either $v_i > v_j$ (the sign of e_{ij} is "-") or $v_j < v_i$ (the sign of e_{ij} is "+") is right considering the status relation between v_i and v_j. The reason is that if the transitivity property does not hold, it is unable to derive the sign for the third edge merely using status theory.

From Table 2, it is also worthy to note that, among the 16 triads, only 4 of them will make contradictory predictions based on two theories. We further examine the percentage of triads satisfying balance and/or status theory on large scale online social networks. The results are shown in Table 3, where "Consistency", "Only Balance", and "Only Status" denotes both theories, or only one balance or status theory are satisfied, respectively, and "No Match" refers no theory satisfied.

From Table 3, it is clear that only a very small fraction of triangles satisfy neither of two theories, i.e., less than 2% on Slashdot and Epinions, and about 7% on Wikirfa. In addition, the consistency between balance and status theories dominates the real world signed directed networks. Hence it is possible to combine the balance and status theory in a joint framework despite the small fraction of contradictory cases. More importantly, while balance theory models the tight relationship among three vertices, the transitivity property of status theory may capture the relationship between every two vertices along a long path. This indicates that two theories may complement each other. We will later show how this characteristic can be successfully exploited in our proposed framework.

Status theory for "bridge" edges. Triads inherently reflects the property of balance theory and the transitivity of status theory. However, as we illustrated in the introduction section, there are a good number of "bridge" edges in social networks which are not included in any triangles. Balance theory does not apply to the nodes involved in this type of edges. Status theory, on the other hand, has a more general application than balance theory as its target is to explain the local patterns of signed links between two adjacent users. As long as a user j has higher status or skills than another user i, there should be a positive out link from i to j. Hence we adopt status theory to learn effective representations which can be used for predicting the sign of "bridge" edges.

3.2 Modeling "Triangle" Edges based on Balance and/or Status Theory

Since the balance and status theories could fit into different triangles in signed directed networks, we let our model learn from the true triangles to capture the latent distribution. We take the triad T_6 in Figure 2 as an example to show the detail of the training process. Given that the sign of e_{ik} is "-" and that of e_{kj} is "-", assume the the ground truth sign of e_{ij} is "+" which follows the principle of "the enemy of my enemy is my friend" in balance theory. Mathematically, taking the sign of three edges together, our goal is to maximize the following objective function.

$$J^{tri}_{T_6} = P(+|e_{ij}) * P(-|e_{ik}) * P(-|e_{kj}), \tag{3}$$

where $J^{tri}_{T_6}$ is the overall probability for predicting the sign of three edges in the triad T_6. On the other hand, if the ground truth sign for e_{ij} is "-" which follows the status theory, we can simply replace the $P(+|e_{ij})$ in Eq. 3 with $P(-|e_{ij})$ when training the model.

We then take all triangles in the network into consideration, maximum the likelihood and define the objective function J^{tri} for all triangles based on balance and/or status theory as follows.

$$J^{tri} = \prod_{t \in T_{sam}} J^{tri}(t), \tag{4}$$

where T_{sam} is the set of sampled triangles. In order to maximize J^{tri}, we use a log loss function L_{tri} to measure the difference between the observation and the prediction, and define L_{tri} as:

$$L_{tri} = -logJ^{tri} = \sum_{t \in T_{sam}} -logJ^{tri}(t) = \sum_{t \in T_{sam}} L^{tri}(t), \tag{5}$$

Since a triangle is constructed from three vertices (i, j, k), we further have:

$$L^{tri}(t) = L^{tri}_{ij} + L^{tri}_{ik(ki)} + L^{tri}_{jk(kj)}, \tag{6}$$

Table 2: Balance and status theory in triangles

	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}	T_{16}
Balance Theory	+	−	+	−	−	+	−	+	+	−	+	−	−	+	−	+
Status Theory	+	+/−	+/−	+	+/−	−	−	+/−	+/−	−	−	+/−	+	+/−	+/−	+

Table 3: The percentage of triads satisfying balance and/or status theory

Dataset	Slashdot	Epinions	Wikirfa
Consistency	75.05%	76.82%	67.74%
Only Balance	16.62%	15.58%	5.88%
Only Status	6.58%	6.59%	19.25%
No Match	1.75%	1.02%	7.13%

where L_{ij}^{tri} is used to measure the difference between the predicted value $P(+|e_{ij})$ and the ground truth value y_{ij} for the sign of the edge e_{ij}, and it can be defined using a cross-entropy loss function. Hence we have:

$$L_{ij}^{tri} = -y_{ij} \log P(+|e_{ij}) - (1 - y_{ij}) \log(1 - P(+|e_{ij})) \tag{7}$$

Similarly we can define the loss for other edges and get $L_{ik(ki)}^{tri}$, $L_{jk(kj)}^{tri}$ in Eq. 6.

3.3 Modeling "Bridge" Edges based on Status Theory

As we illustrated in the previous section, using triangles to model edges could be effective but there are a number of "bridge" edges in real-world signed directed networks. In order to avoid the deterioration of the performance caused by the exclusion of "bridge" edges, we model these edges using status theory from another point of view.

We take a "bridge" edge $a \to b$ as an example. We denote the status score of node v_a, v_b as S_a, S_b. Given that the sign of e_{ab} is "+", we have the status relationship of $S_a < S_b$ if we follow the rule of "the person respected by me should have higher status than me". Mathematically, taking the status relationship into consideration, our goal is to maximize the following objective function:

$$J^s(a, b) = P(Q(a, b)|S_a, S_b), \tag{8}$$

where $J^s(a, b)$ is the probability of status relationship as $S_a < S_b$, and Q is the ground truth of status relationship between a and b defined as:

$$Q(a, b) = \begin{cases} 1(S_a < S_b), & a \to b : + \\ 0(S_a > S_b), & a \to b : - \end{cases} \tag{9}$$

Taking all the "bridge" edges in the network into consideration, we can define the following objective function J_{sta} for these edges.

$$J_{sta} = \prod_{e_{ab} \in E_{bri}} J_{ab}^s, \tag{10}$$

where E_{bri} is the set of "bridge" edges in a signed directed network.

In order to maximize J_{sta}, we use a log loss function L_{sta} to measure the difference between the observation and the prediction, and define L_{sta} as:

$$L_{sta} = -log J_{sta} = \sum_{e_{ab} \in E_{bri}} L_{ab}^s. \tag{11}$$

L_{ab}^s in Eq. 11 is the cross-entropy loss for e_{ab} defined as:

$$\begin{aligned} L_{ab}^s = &- Q(a, b) log P(Q(a, b)|\sigma(-S_a + S_b)) \\ &- (1 - Q(a, b)) log(1 - P(Q(a, b)|\sigma(-S_a + S_b))), \end{aligned} \tag{12}$$

where we use $\sigma(-S_a + S_b)$ as the probability of the status relationship $S_a < S_b$, and σ is a sigmoid function to normalize the status difference value $-S_a + S_b$ to the range of $(0, 1)$.

3.4 BESIDE Model

Based on the mathematically modeled triangles and "bridge" edges, we now combine balance and status theory together and propose a bridge enhanced signed directed network embedding (BESIDE) model. The entire objective function can be written as:

$$L_{all} = L_{tri} + L_{sta} + \lambda_{reg} L_{reg} \tag{13}$$

where $L_{reg} = ||\Theta||_2^2$ is the L_2 regularizer for all weight parameters Θ in the neural network and λ_{reg} is the corresponding weighing factor ($\lambda_{reg} = 0.0001$ in our experiments). With the objective function L_{tri} and L_{sta} given in Eq.5 and Eq.11, our task is to find a function f to measure the probability $P(+|e_{ij})$ or $P(-|e_{ij})$ of the sign of an edge e_{ij} and two functions g_{src} and g_{tar} to get the status score S_i^{src}, S_j^{tar} for source node v_i and target node v_j, respectively. Motivated by recent advances in deep learning which has been proven to be powerful in learning nonlinear representations, we design a novel deep neural network which optimizes the objective function in Eq. 13 and learns the embedding of nodes and edges as well as the function f, g_{src}, g_{tar}. Figure 3 shows the architecture of our model.

The architecture in Figure 3 consists of two components, i.e., the BESIDE_tri component modeling triangles using balance and/or status theory and the BESIDE_sta component modeling "bridge" edges using status theory. The input to BESIDE_tri is a set of triplets $(\mathbf{x_i}, \mathbf{x_j}, \mathbf{x_k})$ denoting the embedding of nodes in triads extracted from the signed directed network. Starting from the node embedding layer of a triad, the BESIDE_tri component aims to optimize the relationships in a triangle in next layers. More formally, we define the probability $P(+|e_{ij})$ of the sign of an edge e_{ij} as:

$$P(+|e_{ij}) = f(\mathbf{x_i}, \mathbf{x_j}) = f(\mathbf{e_{ij}}) = \sigma(\mathbf{e_{ij}} \mathbf{W_1} + \mathbf{b_1}), \tag{14}$$

where σ is a sigmoid function, $\mathbf{W_1}$ and $\mathbf{b_1}$ is the weight and bias in the third layer, respectively, and $\mathbf{e_{ij}}$ is the embedding of an edge e_{ij} and defined as:

$$\mathbf{e_{ij}} = \mathbf{x_i} \mathbf{W_{eh}} + \mathbf{x_j} \mathbf{W_{et}} + \mathbf{b_e}, \tag{15}$$

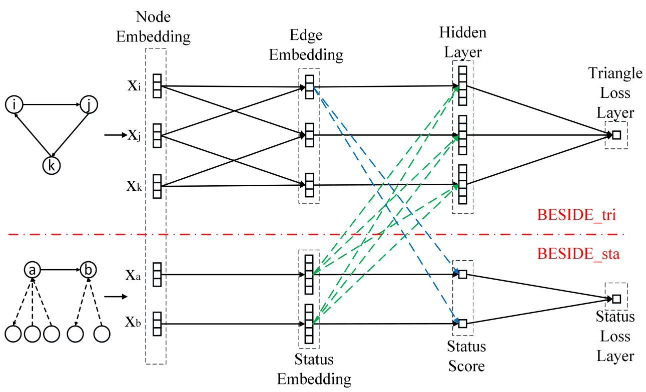


Figure 3: Architecture of BESIDE model. The component above the red dash-dotted horizonal line models triangles while the below one models "bridge" edges. The blue and green dotted lines are interactions between two components.

where $\mathbf{W_{eh}}$ and $\mathbf{W_{et}}$ are the weights and $\mathbf{b_e}$ the bias in the second layer.

The input to BESIDE_sta is a set of tuples $(\mathbf{x_a}, \mathbf{x_b})$ denoting the embedding of two nodes of a "bridge" edge, and the objective is to optimize the relationships between two nodes adjacent to a bridge edge based on a status score. Formally, we define the following status score function g_{src}, g_{tar} for the node v_a and v_b of a "bridge" edge $a \longrightarrow b$ as:

$$\begin{aligned} \mathrm{S_a^{hid}} &= \mathbf{x_a W_{src}} + \mathbf{b_{src}}, \\ S_a &= g_{src}(v_a) = \sigma(\mathrm{S_a^{hid}} \mathbf{W_3} + \mathbf{b_3}), S_a \in (0,1), \\ \mathrm{S_b^{hid}} &= \mathbf{x_b W_{tar}} + \mathbf{b_{tar}}, \\ S_b &= g_{tar}(v_b) = \sigma(\mathrm{S_b^{hid}} \mathbf{W_3} + \mathbf{b_3}), S_b \in (0,1), \end{aligned} \tag{16}$$

where $\mathbf{W_{src}}$, $\mathbf{b_{src}}$, $\mathbf{W_{tar}}$, $\mathbf{b_{tar}}$, $\mathbf{W_3}$ and $\mathbf{b_3}$ are the weights and biases in the second and third layer. $\mathrm{S_a^{hid}}$ and $\mathrm{S_b^{hid}}$ are the status embedding for the node v_a and v_b. S_a and S_b are the status score for the source and target node v_a and v_b, respectively.

To make two components complement each other, we not only share the node embeddings in the first layer, but also make extra interactions between them (shown as the blue and green dotted lines in Figure 3). More formally, for the BESIDE_tri component, we modify Eq. 15 as follows to borrow information from BESIDE_sta and replace $\mathbf{e_{ij}}$ with $\mathbf{e'_{ij}}$:

$$\mathbf{e'_{ij}} = [\mathbf{e_{ij}}; \mathrm{S_i^{hid}} - \mathrm{S_j^{hid}}], \tag{17}$$

where $[;]$ is the concatenation operator of two vectors and $\mathrm{S_i^{hid}} - \mathrm{S_j^{hid}}$ is the status difference vector calculated from Eq. 16.

Similarly, for the BESIDE_sta component, we modify Eq. 16 as follows to borrow information from BESIDE_tri and replace $\mathrm{S_a^{hid}}, \mathrm{S_b^{hid}}$ with $\mathrm{S_a^{hid'}}, \mathrm{S_b^{hid'}}$:

$$\begin{aligned} \mathrm{S_a^{hid'}} &= [\mathrm{S_a^{hid}}; \mathbf{x_a W_{eh}} + \mathbf{b_e}/2] \\ \mathrm{S_b^{hid'}} &= [\mathrm{S_b^{hid}}; \mathbf{x_a W_{et}} + \mathbf{b_e}/2] \end{aligned} \tag{18}$$

where $\mathbf{x_a W_{eh(et)}} + \mathbf{b_e}/2$ (from Eq. 15) is used to enrich the status vector representation for node v_a and v_b, respectively.

3.5 Training BESIDE

To train our BESIDE model, we use mini-batch stochastic gradient descent [2] to update the parameters in the neural network. The training procedure is summarized in Algorithm 1. In line 1 in Algorithm 1, we prepare the input for a mini-batch training based on the sampled triangles and "bridge" edges. To reduce time complexity, we randomly sample one triangle for each edge during each epoch. The "bridge" edges are those which cannot find any triangles, so the input number for the BESIDE component is limited to the number of total "bridge" edges. Thus the total input is restricted to $O(|E|)$. In line 2, we initialize the parameters for our neural network. Then we train the target of our model in forward propagation. We calculate the BESIDE_tri loss L_{tri} in Eq. 6 and use stochastic gradient descent in back propagation to update $\mathbf{X}$ and other relevant parameters $\mathbf{\Omega}$ in the neural network from line 4 to line 8. For the BESIDE_sta component, we have a similar procedure from line 9 to line 13.

Given the learnt embeddings for nodes and the weight and bias parameters, BESIDE_tri component can provide us with edge embeddings from the second layer for sign prediction task, and BESIDE_sta component generates status scores from the third layer for node ranking task, thus the whole BESIDE model could deal with two tasks at the same time.

Algorithm 1 BESIDE Algorithm

Require: signed directed network $G = \{V, E\}$
Ensure: representations for nodes $\mathbf{X}$, status scores for nodes $\mathrm{S_{src}}, \mathrm{S_{tar}}$, the relevant weight and bias parameters $\mathbf{\Omega}$ in the neural network

```
1:  Prepare triangle samples T_sam (from Eq. 4) and "bridge" edges set E_bri (from Eq. 10)
2:  Initialize the parameters of neural network
3:  repeat
4:      for each mini-batch from T_sam do
5:          Forward propagation, calculate L_tri
6:          minimize model loss L_tri, calculate the gradients
7:          Back propagation
8:          Update the relevant parameters
9:      for each mini-batch from E_bri do
10:         Forward propagation, calculate L_sta
11:         minimize model loss L_sta, calculate the gradients
12:         Back propagation
13:         Update the relevant parameters
14: until Convergence
15: return X, S_src, S_tar, Ω
```

4 EXPERIMENTAL EVALUATION ON SIGN PREDICTION

We first conduct sign prediction experiments to check whether our model improves the performance of signed network analysis [23].

4.1 Datasets

We conduct experiments on three well known and publicly available signed social network datasets. Slashdot [14] is a technology-related news website known for its specific user community. Users are allowed to tag each other as friends or foes. Epinions [14] is an online social network of a general consumer review site. Members of the site can decide whether to "trust" each other. Wikirfa [24] records the voting process during "request for adminship(RfA)", where any community member can cast a supporting, neutral, or opposing vote for a Wikipedia editor. We discard neural votes and construct a signed network as [24] did. The statistics of three datasets are summarized in Table 4.

Table 4: The statistics for datasets

Dataset	Node	Edge	+Edge(%)	-Edge(%)
Slashdot	82,140	549,202	77.40	22.60
Epinions	131,828	841,372	85.30	14.70
Wikirfa	11,258	179,418	77.92	22.08

4.2 Baselines and Settings

For this prediction experiment, we use a number of the state-of-the-art baselines and one variant of our method.

DeepWalk(DW) [17] is a network embedding method based on language model skip-gram. Since this method cannot distinguish between positive and negative edges, all the train edges are actually seen as positive edges to construct the graph for it to do the random walk. In case that some nodes have no edges in the training set, we add a self-loop edge for each of them so that all nodes can get their embeddings.

LINE [20] is a network embedding method considering first-order and second-order proximity. It cannot handle negative weights on edges, so we do preprocessing similarly to that in DeepWalk.

Node2Vec(N2V) [4] is also a network embedding methods for unsigned network. It uses two hyper-parameters p and q to generate random walks with the idea of breadth-first search and depth-first search. We use its recommended parameters and tune the p and q follow its paper's steps. Other preprocessing steps are similar to that in DeepWalk.

FExtra [13] is a feature-engineering method in view of balance and status theory. It adopts 7 degree based features in conjunction with 16 social theories based features to model the signed social network. Since FExtra differs from the network embedding methods, we follow the same approach in [13] to obtain the edge features.

SNE [27] is a signed network embedding method which utilizes log-bilinear model with random walk sampling. However, it does not exploit any social theories for signed networks.

SiNE [23] is a signed network embedding method focused on network structure information. It assumes "users should sit closer to their friends (or users with positive links) than their foes" as the balance theory and uses a deep learning framework to optimize this relation. Since this method is developed for undirected network, we follow its sampling method and adjust it to generate batches on three directed signed networks without changing the neural network model.

SIDE [10] is a signed network embedding method based on random walk. It aggregates signs and directions along the path according to balance theory and devise a general likelihood formulation for signed directed connections.

Sign2Vec [7] is similar to SIDE except that it adds a new targeted node sampling strategy to maintain structural balance in higher-order neighborhoods.

BESIDE_tri is a component (above the red line) of our proposed BESIDE framework. It is used to model the triangles in the network.

We first use node representations to compose edge representations. For DeepWalk, LINE, N2V and SiNE, we use their node embeddings as their node features. Then we follow the method in [4] and [7] to get edge features through five operators (average, Hadamard, weighted-L1, weighted-L2, concatenate) and report their best results in this paper. For SNE, it differentiates the source and target node embeddings when constructing edge features. The node features for SiNE consists of only node embeddings, and SIDE adds extra bias terms. The node features for Sign2Vec are made up of node vectors and context vectors. The rest steps to get edge features for SiNE, SIDE and Sign2Vec are the same with DeepWalk. We evaluate all methods except FExtra using source codes provided by their authors. For our BESIDE_tri and BESIDE, we directly use edge embeddings as defined in Eq. 15.

With the learnt edge features, we then train a logistic regression classifier on training set and use it to predict the edge sign in test set. We randomly select 80% edges as training set and the remaining 20% as the test set. We run with different train-test splits for 5 times to get the average scores. For a fair comparison, we set all the node embedding dimension to 20 which is as same as that in SiNE [23] and FExtra [13]. For other parameters in baselines, we follow the recommended settings in their original papers.

4.3 Results for Sign Prediction

We report the average auc, macro-F1, micro-F1 and binary-F1 as evaluation metrics as those in [7, 13, 23]. Table 5 shows the results. Scores in bold denote the highest performance among all methods, and scores with underlines are the highest among all baselines, i.e., those except our BESIDE_tri and BESIDE.

We can observe that our BESIDE model achieves the best performance on three datasets. Our BESIDE_tri component already performs pretty well. Since most of triangles satisfy both balance and status theories, the embeddings of these triangles already contain effective information. When "bridge" edges are incorporated into the model, BESIDE could get better performance than BESIDE_tri.

Three unsign network embedding baselines (DeepWalk, LINE and Node2Vec) are the worst, showing that it is not suitable to apply unsigned network embedding methods to this problem.

Among four signed network embedding baselines (SNE, SiNE, SIDE and Sign2Vec), SNE is not doing well compared to other three. This can be due to its ignorance of sociopsychological theories. Sign2Vec generally gets the best performance among these four baselines. However, it is significantly worse than our BESIDE model, especially on macro-F1. The performance of SIDE is not as good as that in [10] (We also try to use the original dimensionality (128), but it does not change much). This can be due to the fact that the results in [10] are reported on ideal datasets which preprocess 1 degree and 0 in/out degree nodes, and some of these nodes are ignored

Table 5: Results for sign prediction

dataset	metric	DW	LINE	N2V	SNE	SiNE	SIDE	Sign2Vec	FExtra	BESIDE_tri	BESIDE
Slashdot	auc	0.7743	0.5579	0.6318	0.6670	0.8581	0.8495	0.8805	0.8867	0.8759	**0.9092**
	macro-F1	0.5994	0.4368	0.4367	0.4993	0.7512	0.7433	0.7656	0.7390	0.7594	**0.7985**
	micro-F1	0.7779	0.7740	0.7741	0.7785	0.8399	0.8407	0.8438	0.8465	0.8457	**0.8649**
	binary-F1	0.8668	0.8726	0.8727	0.8732	0.8998	0.9015	0.9010	0.9065	0.9035	**0.9142**
Epinions	auc	0.8170	0.6012	0.7484	0.8424	0.9000	0.8730	0.9182	**0.9446**	0.9304	0.9439
	macro-F1	0.6141	0.4738	0.6265	0.7901	0.8294	0.8223	0.8295	0.8069	0.8478	**0.8679**
	micro-F1	0.8693	0.8543	0.8754	0.9123	0.9232	0.9234	0.9210	0.9213	0.9306	**0.9376**
	binary-F1	0.9279	0.9213	0.9314	0.9503	0.9559	0.9564	0.9544	0.9555	0.9600	**0.9638**
Wikirfa	auc	0.7337	0.5881	0.6233	0.6909	0.8631	0.7939	0.8765	0.8597	0.8943	**0.8976**
	macro-F1	0.5635	0.4380	0.5200	0.5241	0.7306	0.6801	0.7497	0.7183	0.7678	**0.7723**
	micro-F1	0.7890	0.7795	0.7892	0.7855	0.8329	0.8201	0.8438	0.8323	0.8530	**0.8553**
	binary-F1	0.8773	0.8761	0.8795	0.8768	0.8966	0.8917	0.9032	0.8975	0.9084	**0.9097**

by SIDE during training. In contrast, all other methods (including ours) use full datasets and learn representations for all nodes. This is practically important. For example, in Wikirfa, some voters (1 degree nodes) who vote once are ignored. This will change the original network structure and reduce the valid voting information (the number of signed directed edges) to the votees.

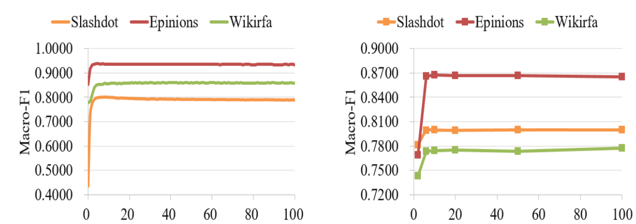
The feature-engineering baseline FExtra shows good performance when compared with others. The reason may be that its low-dimension features are carefully designed based on social theories. Note that FExtra outperforms SiNE in our experiments since our datasets are different from those in [23] although they have same names of datasets. FExtra could beat our BESIDE_tri component in several cases. However, our BESIDE can still have better results overall due to the enhanced information from "bridge" edges.

Since edge signs in above three networks are overwhelmingly positive, a totally positive prediction may yield a 80% or so correct rate. In order to investigate the robustness of our model, we follow the methodology in [5, 13] to create a balanced dataset with equal numbers of positive and negative edges and conduct the sign prediction task on this balanced dataset. The results are shown in Table 6.

From Table 6, we find an overall increasing value for macro-F1. This is reasonable due to the very nature of balanced dataset which favors a metric like macro-F1. All the micro-F1, binary-F1, and auc values decrease on this dataset. However, compared with baselines, the change ratio of our BESIDE model is the smallest. For example, the auc value of FExtra on Epinions drops from 0.9446 to 0.9196, showing a 2.64% decrease. In contrast, the auc value of our model for the full and balanced Epinions dataset is 0.9439 and 0.9437, respectively, almost no change. This clearly proves that the results for our BESIDE model are much more robust than baselines whether we use the full or balanced dataset.

4.4 Parameter Analysis

In this subsection, we investigate the effects of two main hyper-parameters, i.e., the number of iteration *epoch* and the embedding dimension d. We select 80% training edges and 20% test edges as previous subsection does. We set $epoch = 10$, $d = 20$ as default values and change one parameter while keeping the other fixed. Figure 4 shows the results.



(a) *epoch* **(b)** d

Figure 4: Parameter analysis for sign prediction

As shown in Figure 4(a), we could see our method converges quickly and does not fluctuate a lot when *epoch* increases. In Figure 4(b), the performance increases first and becomes almost steady with the increasing number of dimension d (the numeric change of macro-F1 is within 1%). Even with 6 embedding dimension, our model has already achieved pretty good performance. In general, our method is not sensitive to the hyper-parameters and provides robust results for the sign prediction task.

5 EXPERIMENTAL EVALUATION ON NODE RANKING

In this section, we investigate the properties of our status scores $\mathbf{S_{src}}$, $\mathbf{S_{tar}}$ from BESIDE model. Based on status scores, we can further get the global ranking for all nodes in the network. Thus we turn to conduct two types of experiments. Firstly, we apply our status scores on real-world datasets and compare the status of two linked

Table 6: Results for sign prediction on balanced dataset

dataset	metric	DW	LINE	N2V	SNE	SiNE	SIDE	Sign2Vec	FExtra	BESIDE_tri	BESIDE
Slashdot	auc	0.7374	0.5590	0.6529	0.6278	0.8386	0.8415	0.8728	<u>0.8775</u>	0.8398	**0.9049**
	macro-F1	0.6893	0.5393	0.6115	0.5870	0.7747	0.7755	<u>0.7932</u>	0.7929	0.7671	**0.8248**
	micro-F1	0.6908	0.5393	0.6129	0.5870	0.7748	0.7755	0.7933	<u>0.7944</u>	0.7673	**0.8248**
	binary-F1	0.6681	0.5430	0.5878	0.5829	0.7780	0.7764	0.7892	<u>0.8106</u>	0.7742	**0.8246**
Epinions	auc	0.7145	0.5924	0.7442	0.8129	0.8918	0.8673	<u>0.9227</u>	0.9196	0.9109	**0.9437**
	macro-F1	0.6915	0.5664	0.6875	0.7404	0.8229	0.8180	<u>0.8497</u>	0.8366	0.8338	**0.8740**
	micro-F1	0.6921	0.5679	0.6890	0.7405	0.8229	0.8183	<u>0.8497</u>	0.8380	0.8342	**0.8741**
	binary-F1	0.6773	0.5910	0.7093	0.7461	0.8217	0.8255	0.8495	<u>0.8517</u>	0.8417	**0.8765**
Wikirfa	auc	0.6672	0.5808	0.6287	0.6303	0.8466	0.7715	<u>0.8701</u>	0.8374	0.8668	**0.8814**
	macro-F1	0.6204	0.5574	0.5789	0.5962	0.7772	0.7170	<u>0.7896</u>	0.7815	0.7844	**0.8013**
	micro-F1	0.6205	0.5575	0.5915	0.5962	0.7772	0.7172	<u>0.7897</u>	0.7815	0.7845	**0.8013**
	binary-F1	0.6152	0.5573	0.6514	0.5941	0.7756	0.7236	<u>0.7876</u>	0.7808	0.7826	**0.7987**

nodes. Secondly, we compute the global ranking value using status scores and examine its application in an online voting network.

5.1 Baselines

For this ranking experiment, we use the following six baselines and one variant of our BESIDE model.

PageRank [3] is a classic algorithm designed for ranking nodes in unsigned networks. We apply it to the positive subgraph $G+$ to obtain the global ranking values as [25] does.

MPR [18] modifies PageRank [3] by applying it separately on positive subgraph $G+$ and negative subgraph $G-$. Then the final reputation score is computed as $r_i^+ - r_i^-$, where r_i^+ and r_i^- are ranking scores calculated from the previous step.

MHITS [25] is a modified version of HITS [12]. Similar to that in [25], we run HITS on $G+$ and $G-$ separately and combine each node's authority value $a_i^+ - a_i^-$ as its node representation.

Prestige [28] simply combines positive and negative incoming links to give each node a prestige value. If a node receives many positive/negative incoming links, it should have a high/low prestige.

Exp [21] is based on an exponential variation of the PageRank. It assumes a node with a negative reputation is partially trustworthy.

Troll-Trust [25] uses a Bernoulli distribution to characterize each user as either being trustworthy, or being a troll, and constructs a probabilistic model in terms of the links between various users.

BESIDE_sta is a component (below the red line) of our proposed BESIDE framework. It focuses on status theory to model the "bridge" edges.

5.2 Status Comparison on Three Real-World Networks

According to the status theory in signed directed network, the status score of each node can be interpreted as "the person respected by me should have higher status than me". To derive insights into the property of such status scores in real-world signed social network, we design a status comparison experiment: we take the test edges as the ground truth, and compare the status between two adjacent nodes by their status scores. For example, a positive(negative) edge $v_i \rightarrow v_j$ can be transformed into a status comparison $S_i < (>) S_j$ based on status theory. In this way, we can measure how the status scores generated by different methods are consistent with the ground truth.

We conduct the status comparison experiments on the Slashdot, Epinions and Wikirfa datasets. We use 80% edges as training set and 20% edges as test set and use accuracy as evaluation metric follows [18, 25]. For Troll-Trust, we try different combinations of $\beta = [0.01, 0.1, 0.2, 0.5, 0.9]$ and $\lambda_1 = [0.1, 0.5, 1.0, 5.0, 10.0, 100.0]$ and choose the best results for Slashdot, Epinions and Wikirfa, respectively. For other baselines, we follow the settings in Troll-Trust [25]. For our BESIDE model, we simply get the status scores $\mathbf{S_{src}}, \mathbf{S_{tar}}$ from the model trained in the sign prediction task and apply them to this task to see whether $\mathbf{S_{src}}(i) < (>) \mathbf{S_{tar}}(j)$ satisfies positive(negative) $v_i \rightarrow v_j$ relation. The results are shown in Table 7.

BESIDE well captures the status property and it achieves the best performance among all methods on three networks. This could be attributed to that our BESIDE is motivated by social-psychological theory and thus the status scores agree fairly well with the human behaviors in real-life networks. Another interesting finding would be the significant improvements of BESIDE over BESIDE_sta on Epinions and Wikirfa, which proves that the BESIDE_tri component in our framework provides the BESIDE_sta component with additional information and helps enhance its performance. We also observe a small raise on Slashdot. The reason may be that Slashdot contains nearly half of bridge-edges, and it is hard for BESIDE_sta to benefit from BESIDE_tri on this dataset.

5.3 Global Ranking on Wikirfa Network

In order to investigate whether our status score could help find something specific to signed directed networks, we analyse the top 10 nodes ranked by various methods. For all baselines which are originally designed for the node ranking task, the nodes in the

Table 7: Accuracy for status comparison on Slashdot, Epinions and Wikirfa

dataset \ method	Prestige	PageRank	Exp	MPR	MHITS	Troll-Trust	BESIDE_sta	BESIDE
Slashdot	0.4619	0.6273	0.5920	0.5815	0.5518	0.5915	0.8595	**0.8611**
Epinions	0.5134	0.6515	0.6457	0.6503	0.5883	0.6424	0.8559	**0.9131**
Wikirfa	0.6397	0.6629	0.6770	0.6744	0.6363	0.6783	0.7779	**0.8276**

network can be directly ranked using their ranking scores. Since our proposed BESIDE differentiates the positive and negative status scores, it is necessary to combine them together to get an integrated ranking score. To this end, we refer to the definition of PageRank [3] and propose the global ranking score for BESIDE as the combination of positive and negative score normalized by the number of out links from the source node. The definition is as follows:

$$R_i = \sum_{j \in In_+(i)} \frac{S'_{src}(j)}{|Out(j)|} - \sum_{j \in In_-(i)} \frac{S'_{src}(j)}{|Out(j)|} \qquad (19)$$

where R_i is the ranking score of node v_i, $In_+(i)$ and $In_-(i)$ denoting the nodes pointing positively or negatively to node v_i, $|Out(j)|$ denoting the out degree of node v_j, and $S'_{src}(j)$ is the normalized source status score of node v_j in the range of $[0, 1]$.

We conduct the ranking experiment on Wikirfa network because it is the only dataset where the ranking of nodes can be implicitly reflected by the voting results (being elected as an administrator or not). This dataset also contains textual contents which can provide insightful analysis. During the RfA (request for adminship) process initiated in 2003 through May 2013, there were 189,004 distinct voter/votee pairs among which 3,494 users ran for elections. Among these users, 1,591 ones were never elected, and 1,885 and 18 users were elected once and twice, respectively.

Intuitively, the win/loss of the election reflects a user's status of being trusted, and 18 twice-elected users should have the highest status while 1,591 none-elected users should be the lowest. We run different methods on Wikirfa and show twice, once, and none elected users in Table 8. We adopt the conventional top-10 metric in information retrieval and recommender systems to present results.

In Table 8, the green (also in italic), black, and blue (also in bold) colored users were never elected, elected once, or elected twice, respectively. Note that the results for Prestige are not presented because it cannot distinguish its top nodes. All nodes with only positive in-links will get the same top ranking scores 1.0, and thus Prestige is not suitable for this task at all. We have the following important notes for Table 8.

- PageRank selects "ProtectionBot" into the top-10 list as this user has many in-links (> 150 support votes in training data). However, this user has never been elected as the administrator during the entire period. This indicates that the method like PageRank which does not take negative edges into consideration may make a mistake for the node ranking task in signed directed networks.
- Troll-Trust is the second worst since all users in its top-10 list are elected only once. A close look at the selected top-10 users reveals that they do not have any negative in-links. This indicates that Troll-Trust is too strict to include nodes with negative links.
- MPR and MHITS are similar. Nine users are elected as administrators once and one user is elected twice. These two methods model negative links to some extent, but positive in-links still play the leading role. The average number of negative in-links in training data for top-10 users is 7.0 and 1.9 in MPR and MHITS, respectively, while that for the positive ones is over 100.0 for both methods.
- Exp finds two twice-elected users. The average number of negative in-links for the top 10 users for Exp is higher than that for MPR and MHITS. This infers that the inclusion of more negative links can be helpful in finding more twice-elected users.
- BESIDE achieves the best performance in that it selects the most twice-elected users among all methods. The twice-elected users "HJ Mitchell" and "PeterSymonds" partially overlap with MPR, MHITS, or Exp. In addition, BESIDE finds an extra node "Everyking": a very special user with more than 300 positive in-links as well as 200 negative ones but still being elected twice.

In looking at why "Everyking" can be elected twice, we find that this is a highly controversial user. During his first election, voters are predominantly in support for him. They thought he was "very active", and "filling important gaps with good articles, especially on African politcs". However, the second win of his election was dramatic. Many opponents criticized him as "wrote a lot of cruft articles" and "overblown edit war on [[Ashlee Simpson]]". These opponents formed negative links to "Everyking". However, the firm supporters thought he was "One of the most prolific and productive contributors in the history of the project" and "who is clearly working to make the encyclopedia better". Meanwhile, some of supporters argued that "committee found no actual misuse of the tools" and "contributed well later". The opponents seemed overwhelmed by his supporters and "Everyking" was elected successfully.

In summary, our proposed BESIDE model computes ranking scores based on status theory, leveraging both negative and positive links. This allows for finding more twice-elected users than other methods. Moreover, in Table 8, we observe most of once-elected users in BESIDE overlap with those in different approaches. It has 5 once-elected users in common with Exp, and 2 and 1 in common with Troll-Trust and MHITS, respectively. The consistency in once-elected users further proves the soundness of our BESIDE model.

Table 8: Global ranking results on Wikirfa. Green-colored (also in italic) users were never elected as administrator, black-colored users were elected once and blue-colored (also in bold) users were elected twice.

rank \ method	PageRank	Exp	MPR	MHITS	Troll-Trust	BESIDE
1	West.andrew.g	Werdna	Anomie	**PeterSymonds**	SarahStierch	Can't sleep...
2	Cobi	Can't sleep...	Legoktm	Carcharoth	Dabomb87	SarahStierch
3	*ProtectionBot*	**PeterSymonds**	Mkdw	Newyorkbrad	Soap	Phaedriel
4	Anomie	Phaedriel	West.andrew.g	Jake Wartenberg	BD2412	DerHexer
5	Jason Quinn	Newyorkbrad	Tom Morris	Jbmurray	Persian Poet Gal	Alex Bakharev
6	RedirectCleanupBot	Cobi	Arsenikk	Tinucherian	Boing! said Zebedee	Werdna
7	lustiger seth	Drmies	**HJ Mitchell**	Scarian	Jake Wartenberg	**HJ Mitchell**
8	Dinoguy1000	Czrzussian	MGA73	John Carter	Carcharoth	**Everyking**
9	Bellhalla	SarahStierch	Drmies	Soap	Berean Hunter	Dabomb87
10	TommyBoy	**Nev1**	Rambo's Revenge	Phaedriel	John Carter	**PeterSymonds**

6 CONCLUSION

In this paper, we propose a novel BESIDE model to learn effective representations for signed social networks. We first point out that there exist a good number of "bridge" edges that differ from triangles in the network. We then mathematically model "bridge" edges using status theory and model triangles using balance and/or status theory. We next design a novel deep neural structure to combine "bridge" edges and triangles that work in a complementary manner. Based on the deep network, we learn the node embedding and edge embedding denoting the status of a node and the sign of an edge, which can facilitate diverse social computing tasks like status comparison, node ranking, and sign prediction. We conduct extensive experiments on three real-world signed social networks. The results demonstrate that the BESIDE model achieves the state-of-the-art performance, which evinces the need for incorporating "bridge" edges in the signed directed network embedding.

In the future, we plan to investigate how the learnt representations can be used in other applications like community detection or recommendation. We are also interested in making connections between our model and the social properties of real world networks.

ACKNOWLEDGMENTS

The work described in this paper has been supported in part by the NSFC projects (61572376, 91646206), and the 111 project(B07037).

REFERENCES

[1] Smriti Bhagat, Graham Cormode, and S Muthukrishnan. 2011. Node classification in social networks. In *Social network data analytics*. Springer, 115–148.
[2] Léon Bottou. 1991. Stochastic gradient learning in neural networks. *Proc. of Neuro-Nımes* 91, 8 (1991).
[3] Sergey Brin and Lawrence Page. 2012. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer networks* 56, 18 (2012), 3825–3833.
[4] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *Proc. of SIGKDD*. 855–864.
[5] Ramanthan Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. 2004. Propagation of trust and distrust. In *Proc. of WWW*. ACM, 403–412.
[6] Fritz Heider. 1946. Attitudes and cognitive organization. *The Journal of psychology* 21, 1 (1946), 107–112.
[7] Mohammad Raihanul Islam, B Aditya Prakash, and Naren Ramakrishnan. 2018. Distributed Representation of Signed Networks. In *Accepted by PAKDD*.
[8] Jinhong Jung, Woojeong Jin, Lee Sael, and U Kang. 2016. Personalized ranking in signed networks using signed random walk with restart. In *Proc. of ICDM*. IEEE, 973–978.
[9] Cristobald de Kerchove and Paul Van Dooren. 2008. The pagetrust algorithm: How to rank web pages when negative links are allowed?. In *Proc. of SDM*. SIAM, 346–352.
[10] Junghwan Kim, Haekyu Park, Ji-Eun Lee, and U Kang. 2018. SIDE: Representation Learning in Signed Directed Networks. In *Proc. of WWW*. International World Wide Web Conferences Steering Committee, 509–518.
[11] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
[12] Jon M Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *JACM* 46, 5 (1999), 604–632.
[13] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting positive and negative links in online social networks. In *Proc. of WWW*. ACM, 641–650.
[14] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed networks in social media. In *Proc. of SIGCHI*. ACM, 1361–1370.
[15] David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *JASIST* 58, 7 (2007), 1019–1031.
[16] Symeon Papadopoulos, Yiannis Kompatsiaris, Athena Vakali, and Ploutarchos Spyridonos. 2012. Community detection in social media. *DMKD* 24, 3 (2012), 515–554.
[17] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations.. In *Proc. of SIGKDD*. 701–710.
[18] Moshen Shahriari and Mahdi Jalili. 2014. Ranking nodes in signed social networks. *SNAM* 4, 1 (2014), 172.
[19] Jiliang Tang, Yi Chang, Charu Aggarwal, and Huan Liu. 2016. A survey of signed network mining in social media. *ACM Computing Surveys (CSUR)* 49, 3 (2016), 42.
[20] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-scale Information Network Embedding. In *Proc. of WWW*. 1067–1077.
[21] Vincent A Traag, Yurii E Nesterov, and Paul Van Dooren. 2010. Exponential Ranking: Taking into Account Negative Links.. In *SocInfo*. Springer, 192–202.
[22] Suhang Wang, Charu Aggarwal, Jiliang Tang, and Huan Liu. 2017. Attributed signed network embedding. In *Proc. of CIKM*. ACM, 137–146.
[23] Suhang Wang, Jiliang Tang, Charu Aggarwal, Yi Chang, and Huan Liu. 2017. Signed network embedding in social media. In *Proc. of SDM*. SIAM, 327–335.
[24] Robert West, Hristo S Paskov, Jure Leskovec, and Christopher Potts. 2014. Exploiting Social Network Structure for Person-to-Person Sentiment Analysis. *TACL* 2, 1 (2014), 297–310.
[25] Zhaoming Wu, Charu C Aggarwal, and Jimeng Sun. 2016. The troll-trust model for ranking in signed networks. In *Proc. of ICDM*. ACM, 447–456.
[26] Xiaofeng Yu and Junqing Xie. 2014. Learning Interactions for Social Prediction in Large-scale Networks. In *Proc. of CIKM*. 161–170.
[27] Shuhan Yuan, Xintao Wu, and Yang Xiang. 2017. Sne: signed network embedding. In *Proc. of PAKDD*. Springer, 183–195.
[28] Kiyana Zolfaghar and Abdollah Aghaie. 2010. Mining trust and distrust relationships in social Web applications. In *Proc. of ICCP*. IEEE, 73–80.