

Received 20 September 2022, accepted 4 October 2022, date of publication 10 October 2022, date of current version 17 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3213319

## RESEARCH ARTICLE

# Signed Network Node Embedding via Dual Attention Mechanism

ZEKUN LU<sup>1,2</sup>, (Member, IEEE), QIANCHENG YU<sup>1,2</sup>, (Member, IEEE), XIA LI<sup>1</sup>, (Member, IEEE), XIAONING LI<sup>1</sup>, (Member, IEEE), AND AO QIANGWANG<sup>1</sup>, (Member, IEEE)

<sup>1</sup>The College of Computer Science and Engineering, North Minzu University, Ningxia 750002, China

<sup>2</sup>Laboratory of Graphics and Images of the State Ethnic Affairs Commission, Ningxia 750002, China

Corresponding author: Yu Qiancheng (1999019@nmu.edu.cn)

This work was supported in part by the 2022 University Research Platform “Digital Agriculture Empowering Ningxia Rural Revitalization Innovation Team” of Northern University for Nationalities under Grant 2022PT\_S10, in part by the National Natural Science Foundation of China under Grant 62062001, and in part by the Major Key Project of School-Enterprise Joint Innovation in Yinchuan 2022 through the Development and application of Hanas Smart Gas Management Platform and the Innovation Project of North Minzu University under Grant YCX21097.

**ABSTRACT** In signed networks, GNNs are used to get node embedding by aggregating node neighbor information. Most of the existing methods aggregate neighbor information from the node level, and the different paths between nodes and neighbors will also affect node embedding. The target node and its neighbors have different link positive, negative signs and link directions, which together constitute different paths. These different paths have different contributions to the target node. Based on the structural balance theory and status theory, this paper divides the different paths between nodes and their neighbors into 20 kinds of motifs, which are using to capture the different effects of paths on target nodes. Comprehensive consideration at the node level and path level, SNEDA (Signed Network Embedding via dual attention Mechanism) is proposed based on the graph attention Network. The model has two attention mechanisms: node-level attention captures different influences between nodes at the node level; path-level attention captures the different influences between motifs at the path level. The final vector representation of nodes is obtained by aggregating neighbor information selectively based on important motifs, and the vector representation is applied to link prediction. Experiments on four real social network data sets show that the network representation obtained by the model can improve the accuracy of link prediction. Experimental results demonstrate the effectiveness of the proposed framework through a signed link prediction task on four real-world signed network datasets.

**INDEX TERMS** Network embedding, graph neural networks, signed network, graph attention, link prediction.

## I. INTRODUCTION

There are a large number of network structures in the real world, such as social networks, biological protein networks, citation networks, transportation networks, chemical molecular networks, and so on [1]. These network structures are huge and complex, which implies a lot of rich knowledge. Using complex network analysis methods to study these network structures will help people better mine the laws hidden in

The associate editor coordinating the review of this manuscript and approving it for publication was Giacomo Fiumara<sup>1</sup>.

network data [2]. However, the network structure has the characteristics of large data scale and sparse structure, which cannot be directly analyzed by the network analysis method [3]. Network representation learning extends deep learning to network data analysis, aiming to learn low-dimensional vector representations of network nodes while preserving the network topology and node properties [4], [5]. The nodes are represented as low-dimensional, real-valued, dense vector forms so that the obtained vector representation has the ability to represent as well as reason in the vector space [6]. The vector representation can be applied to downstream tasks, such

as link prediction, node classification, community discovery, and personalized recommendation [7], [8].

Signed network is a two-dimensional network containing positive and negative antagonistic relations. Positive relations include positive relations such as a friend, support, and like, and negative relations include negative relations such as enemy, opposition, and dislike [9]. In the process of node embedding in the signed networks, positive edge, negative edge, and edge direction all pose great challenges to node embedding [10], [11]. In this regard, researchers use the special sociological theory of signed networks (structure balance theory and position theory) to model the network structure to correctly distinguish positive and negative edges, and combine it with graph neural networks or deep learning to learn node embedding [12], [13]. This kind of method also achieves optimal performance in the field of signed network embedding. However, these methods focus on the influence of node neighbors from the node level and ignore the feature of path information. In terms of paths, different directions and symbols of links between nodes and their neighbors form complex paths, and these different paths also affect node embedding.

In this paper, the SNEDA method is proposed to capture path information (link different directions and symbols) by dividing different motifs, to learn the influence between nodes under each motif with node-level attention, and to learn the weight between different motifs with path-level attention. The vector representation of nodes is obtained based on the aggregated neighbor information of important motifs. The obtained vector representation integrates node neighbor information and network structure information. The experiment proves that the SNEDA model can improve the quality of node embedding.

The major contributions of this paper are as follows:

- 1) We propose a path-attentional layer, which estimates the importance coefficient for different motifs for the embedding aggregation process.
- 2) We introduce the HAN to model the signed network and design a new motif based GNN model for signed networks named SNEDA.
- 3) We evaluate the effectiveness of the proposed framework SNEDA on several real world signed network datasets through the signed link prediction task.

The rest of this paper is organized as follows. In Section II, related works are given. Section III introduces sociological theory and motifs. Section IV introduces the SNEDA framework. The experimental studies are shown in Section V. Finally, the conclusions are given in Section VI.

## II. RELATED RESEARCH

The methods of network representation learning include methods based on matrix decomposition [14], random walk, and deep learning. DeepWalk [15] and Node2Vec [16] are learning methods based on random walk network representation, which have a good effect in dealing with simple

graph structures. However, the real network is sparse and highly nonlinear. These methods can not capture the structure information of the network, nor retain the local and global information of the network. SDNE [5] model uses an automatic encoder to optimize the first-order and second-order similarity at the same time. The first-order similarity of the model is used as supervised information to retain the local structure of the network; As an unsupervised part, second-order similarity captures the global structure of the network. It is a semi-supervised deep learning model. GCN is a kind of learning node embedding by combining topology and node attribute information in the graph. It belongs to direct push learning, but it can not be directly generalized to nodes that have not appeared in the training process [10], [17]. Therefore, GraphSage proposed that node embedding can aggregate the neighbor information of nodes through a common aggregation function [18]. When training, it only needs to obtain this aggregation function, which can be generalized to unknown nodes. GCN combines the characteristics of nearest neighbor nodes and the structure of the graph, and it is impossible to assign different weights to neighbors during convolution, which limits the generalization ability of the trained model in other network structures [19]. Therefore, the graph attention network GAT is proposed to allocate different attention weights between nodes independently of the graph structure and aggregate the neighbor information with different weights, to greatly improve the expression ability of the graph neural network model [20].

However, the above method is designed for unsigned networks, which have different properties (negative links) compared with unsigned networks. Therefore, the learning method of unsigned network representation can not be directly applied to signed networks.

In recent years, the research on signed network embedding is mainly based on structural balance theory and status theory and uses deep learning technology to represent and learn the network. Shuhan [21] first proposed the SNE model of the signed network with the method of deep learning. SNE combines the edge symbol information and node representation of path nodes with log bilinear model; After that, Shuhan [22] extended the structural balance theory, proposed SiNE model for signed network modeling, and proposed a triple with positive and negative triangular relationship to ensure that the distance between positive relationship node pairs is far less than that between negative relationship node pairs, and gave the measure of similarity, which can better represent the node representation; Tyler derr [23] uses the balance theory to aggregate and propagate the multi-layer information of GCN model in signed networks, but this method ignores the different effects between nodes; After that, Li Yu [24] and others introduced the attention mechanism into the SGCN<sup>1</sup> method, allocated different weight coefficients between node pairs, and aggregated the neighbor information of nodes with the structure balance theory; Huang Junjie [25] and others divided the network structure into different topics and aggregated the node neighbor information under each topic

with gat; Subsequently, Huang Junjie [26] designed a new information dissemination and aggregation mechanism using the structural balance theory, and used an average sampling theory to learn node embedding. Most of the above methods consider the influence between nodes, and different edge directions will also affect the nodes to varying degrees. Therefore, this paper proposes the SNEDA method, which uses a dual attention mechanism to capture the influence between nodes and the influence of edge direction, to make the vector representation of target nodes more rich and complete.

### III. THEORETICAL KNOWLEDGE

#### A. STRUCTURAL BALANCE THEORY

##### 1) STRUCTURALLY BALANCED TRIANGLE

It can be judged by the product of the signs of the three sides of the triangle: if it is positive, the structure of the triangle is balanced; otherwise, the structure is unbalanced. The structural balance [27] of the above triangle is determined from the perspective of sociology and psychology, and can be simply summarized into the following four intuitive understandings: friends of friends are my friends; enemies of friends are my enemies; friends of enemies are my enemies; The enemy of the enemy is my friend. Research shows that in real signed networks, the number of structurally balanced triangles is much larger than that of structurally unbalanced ones, and the unbalanced network gradually evolves into a balanced network over time [28].

##### 2) STRUCTURAL BALANCE CIRCLES

If an  $L$ -circles ( $L \geq 3$ ) contains an even number of negative edges, the structure is balanced, otherwise the structure is unbalanced.

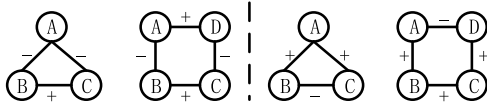


FIGURE 1. Schematic diagram of structural balance (left) and structural imbalance (right).

#### B. SOCIAL STATUS THEORY

Structural balance theory provides a theoretical basis for the analysis of unsigned networks, but there is a large deviation in this theory when it comes to directed signed networks. Subsequently, Leskovec and Kleinbergand [29] proposed a social status theory for signed networks, which holds that if there is a positive edge from  $A$  to  $B$ , then  $A$  has a higher social status than  $B$ . If there is a negative edge from  $A$  to  $B$ , then  $B$  has a higher social status than  $A$ , and this status is transitive.

In a signed network composed of three nodes, the method to determine whether a triangle conforms to the social status theory is as follows: First, reverse the direction of all negative links in the triangle, and convert the signs on the links to positive signs. If the triangle cannot form a cyclic loop, then this triangle conforms to the social status theory, otherwise,

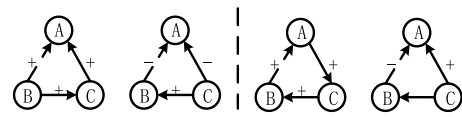


FIGURE 2. Status balance (left) and status imbalance (right).

it does not conform. If each member in a system follows the same position sorting method and there is no conflict in position, then as long as the direction of the edge is known, the sign of the edge can be inferred.

#### C. SIGNED MOTIFS

Discussion based on the above status theory, negative links have different properties from positive links, and different link directions represent different meanings [30], [31]. To distinguish different types of node neighbors, the network structure is divided into different signed motifs to learn respectively according to different directions and symbols of links, as shown in Figure 3. The node is the first-order neighbor of the node, with a total of 4 different motifs. The node is the second-order neighbor of the node, with a total of 16 different motifs.

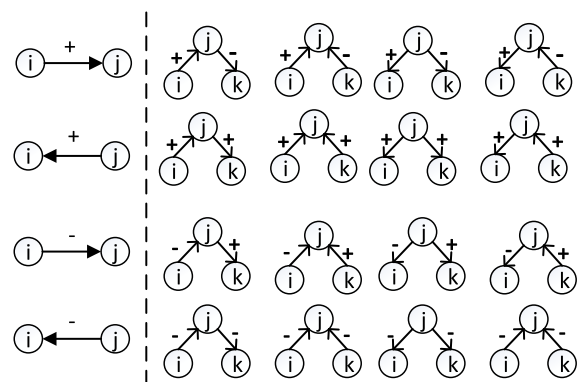


FIGURE 3. Signed motifs.

### IV. MODEL INTRODUCTION

This section will introduce the SNEDA model in detail. As shown in Figure 4, different motifs are generated through different types of node neighbors; Node level attention is used to learn the weight of neighbors based on motif and aggregate them to obtain specific node embedding; The importance of learning different motifs with path level attention; The vector representation of the node is obtained by aggregating the neighbor information based on the important motif; Through the downstream task of link prediction, the embedding quality of model nodes is detected. Several notations are listed in Table 1.

#### A. NODE NEIGHBOR SAMPLING

In real social networks, users' link behavior will be affected by neighbor nodes, and each different type of neighbor node

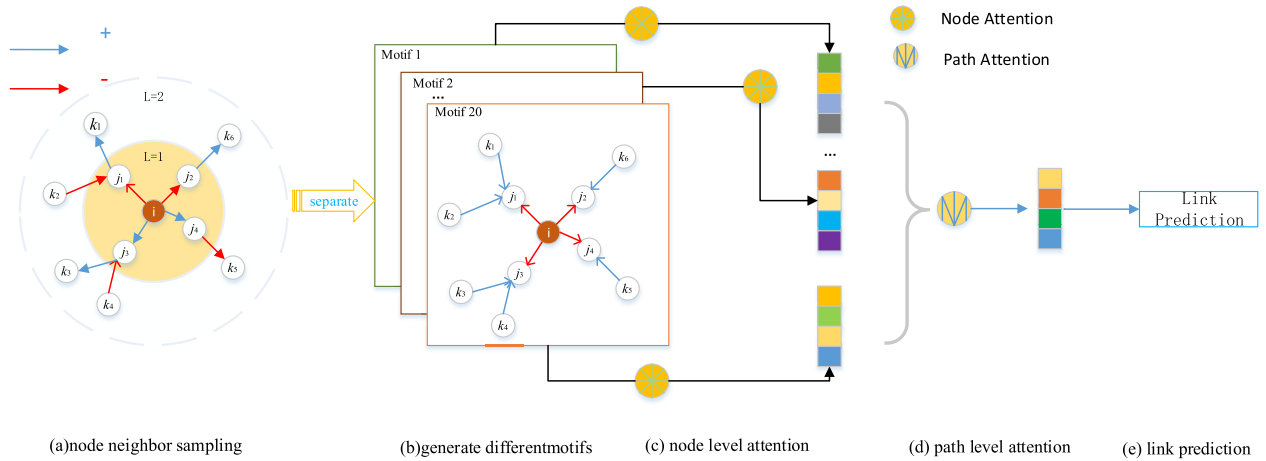


FIGURE 4. An illustration of SNEDA model.

TABLE 1. Notations and explanations.

Notation	Explanation
$F_m(v_i)$	The first order neighbor set of node $v_i$
$S_m(v_i)$	The second order neighbors set of node $v_i$
$\phi_m$	The $m$ kinds of signed motifs, in this paper $m=20$
$a_{ij}^\phi$	The attention coefficient of the node pair $(v_i, v_j)$
$h_i^\phi$	The node $v_i$ vector representation obtained by node-level attention under a certain motif
$d$	The node level attention vector
$q$	The path level attention vector
$B_\phi$	The weight coefficient under a certain motif
$Z_i$	The final target node $v_i$ vector representation obtained by SNEMA model

has a different impact on the target node. To eliminate the different effects caused by different edge directions and link symbols, different motifs are divided to learn the influence between nodes and neighbors under the same motif. Therefore, motifs are selected to sample node neighbors information.

Node neighbor sampling specifically refers to sampling the first-order neighbor and the second-order neighbor of the target node. The first-order neighbor can be divided into four motifs. First order neighbor set  $F = \{F_m(v_i) | v_i \in V, 1 \leq m \leq 4\}$ , represents the first-order neighbor set,  $V$  represents all nodes in the network structure;  $N(v_i) = \cup_{m=1}^4 F_m(v_i)$ , where  $\cup$  represents Union,  $N(v_i)$  represents the set of first-order neighbors. The sampling of  $F$  is mainly based on the idea of a random walk:

- a) starting from the target node  $v_i$ , it swims to the node  $v_j$  in turn, and  $v_j$  is satisfied that  $v_i$  is a neighbor,  $v_j \in N(v_i)$ ;
- b) get the value of  $m$  according to the link direction and symbol between the node  $v_i$  and the node, which will be added to the set  $F_m(v_i)$ .

repeat steps a) and b) until all first-order neighbor nodes have been swimming away.

Similarly, the second-order neighbors can be divided into 16 kinds of motifs,  $S_m(v_i)$  representing the second-order neighbor set, install the super parameter  $p$  to control the number of second-order neighbors. The specific steps of sampling the second-order neighbor sets are as follows:

- c) starting from the target node  $v_i$ , it walks to the node  $v_j$  in turn and satisfies the first-order neighbor of the target node  $v_i$ ;
- d) then start from the node  $v_j$ , swim to the node,  $v_k$  in turn, meet the node's first-order neighbors, determine the motif according to the direction and symbol of the path, and join  $v_k$  into  $S_m(v_i)$ ;

Repeat steps c) and d) until all nodes are completed; After the walk, the nodes in the set are sorted in descending order according to the Degree of Centrality of the nodes, and the nodes before ranking  $P$  are retained, that is, the sampling of the second-order neighbor set  $s$  is completed.

Through node neighbor sampling, the node first-order neighbor set and second-order neighbor set  $S_m(v_i)$  are finally obtained.  $\phi = \{F_m(v_i) \cup S_m(v_i)\}$ ,  $\{\phi_1, \phi_2 \dots \phi_{20}\}$  represent different motifs.

### B. NODE LEVEL ATTENTION

Based on the above discussion, we choose node-level attention to learn the influence between nodes and neighbors. Node level attention can learn the weight coefficients of nodes under the same motif and their neighbors, and aggregate the learned vectors to obtain the vector representation of nodes under a specific motif. This section will introduce how to use node-level attention to learn the impact between node pairs, aggregate these learned neighbor representations, and generate a new embedded representation of the target node.

The self-attention mechanism can learn the weight between each node pair. The importance of different node pairs under different motifs can be expressed:

$$e_{ij}^\phi = \text{att}_{node}(h_i, h_j; \phi) \quad (1)$$

Here  $att_{node}$  represents the deep neural network executing node level attention, and the attention coefficient  $e_{ij}^\phi$  represents the importance of nodes to nodes  $V_i$  under a specific motif;  $h_i, h_j$  represents the eigenvector of the node  $V_i V_j$ ; used  $\phi$  to represent different motifs, that is, motifs can be used to represent  $\{\phi_1, \phi_2 \dots \phi_{20}\}$ .

Then, normalize it with softmax, and the relevant attention coefficient:

$$a_{ij}^\phi = \frac{\exp(\sigma(a_1^T[h_i||h_j]))}{\sum_{k \in N_i^\phi} \exp(\sigma(a_1^T[h_i||h_k]))} \quad (2)$$

Here,  $\sigma$  represents the activation function, the attention coefficient  $a_{ij}^\phi$  of the node pair under the specific motif,  $N_i^\phi$  represents the neighbor set of a node  $V_j$  under the specific motif,  $||$  and  $|||$  both represent the connection operation.

After that, the neighbor information of the sink node  $V_i$  can be aggregated through the feature vector and corresponding coefficient of the neighbor, which is expressed as follows:

$$h_i^\phi = \sigma\left(\sum_{j \in N_i^\phi} a_{ij}^\phi h_j\right) \quad (3)$$

After node-level attention learning, nodes  $v_i$  generate new feature vectors. In a given set  $\{\phi_1, \phi_2 \dots \phi_{20}\}$ , all feature vector sets  $\{h_i^{\phi_1}, h_i^{\phi_2} \dots h_i^{\phi_{20}}\}$  under different motifs of nodes  $v_i$  are finally learned by node-level attention.

### C. PATH LEVEL ATTENTION

In signed networks, each node contains many types of neighbors, and the node embedding of node attention can only reflect the node from one side. Based on the above discussion, node-level attention helps us to better pick out important node neighbor information. To learn more comprehensive node embedding, we need to integrate more abundant path information, Path-level attention helps us choose important paths better. In this way, our model will selectively aggregate neighbor information based on important relationship paths, which enhance the ability to represent features. Note that node-level attention and path-level attention are not specific techniques, but we consider the influence of node neighbors and paths on the target node at the node level and path level and use the attention mechanism to capture this influence. This section will introduce how to use path-level attention to learn the impact of different motifs

Take the node embedding learned by node level attention as the input, and the learning weight of each motif is as follows

$$B_{\phi_1}, B_{\phi_2} \dots B_{\phi_{20}} = att_{path}(h_{\phi_1}, h_{\phi_2} \dots h_{\phi_{20}}) \quad (4)$$

Here  $att_{path}$  represents the depth neural network of execution path attention and represents the weight of the motif.

Embed the nodes learned by node level attention into non-linear changes (for example, one layer MLP), average the importance of all motifs, and the importance of each motif is represented by:

$$w_{\phi_i} = \frac{1}{|V|} \sum_{i \in V} q^T \cdot \tanh(W \cdot h_i^{\phi_m} + b) \quad (5)$$

Here  $W$  is the weight matrix,  $b$  is the deviation vector, and  $q$  is the path level attention vector. All embedding of different motifs shares the above parameters. After getting the importance of each motif, normalize it with the softmax function:

$$B_{\phi_i} = \frac{\exp(\sigma(w_{\phi_i}))}{\sum_{i=1}^m \exp(\sigma(w_{\phi_i}))} \quad (6)$$

Here  $B_{\phi_i}$  represents the weight coefficient of the motif. The higher the weight, the more important the motif is. Different motifs have different contributions to the target nodes. Through learning, the weight coefficients of different motifs are obtained to obtain the final target node embedded  $Z$ :

$$Z_i = \sum_{i=1}^m B_{\phi_i} \cdot h_i^{\phi_i} \quad (7)$$

### D. LINK PREDICTION

Through the experiment of link prediction to verify the embedding quality of model learning nodes, the vector representation of each node is learned through the SNEDA model, and the node vector representation in the training set is input into the binary classification logistic regression model as the node feature for the experiment of link prediction. To better learn the model parameters, SNEDA model adopts Cross Entropy as the loss function, which is defined as follows:

$$Loss = \sum_{v^+ \in N(u)^+} \log(\sigma(Z_u^T Z_{v^+})) - C \sum_{v^- \in N(u)^-} \log(\sigma(-Z_u^T Z_{v^-})) \quad (8)$$

Here  $C$  is used to adjust the proportion of the number of positive connections and negative connections. It is the set of all neighbors of the node's positive links and the set of all neighbors of the node's negative links. The function reflects that the embedding of friends is similar, while the embedding of enemies is not. The model parameters are updated by continuously reducing the loss of cross-entropy. After several optimizations, when the loss tends to be stable, the final vector representation of the node  $V_i$  is obtained. The specific algorithm process is as follows:

### V. EXPERIMENT

In the experiment, all codes were written in PyTorch programming language. The computer is configured with CPU i7-6700, six cores and twelve threads, the memory of,1GB, and a graphics card AMD R7 2GB.

The experimental process is shown in Figure 5. Firstly, the vector representation of nodes is obtained by representing the learning process; 80% of the connected edges are randomly selected as the training set, and these connected edges will produce the vector representation of nodes through the SNEDA model; The remaining 20% of the connected edges are used as the test set, and the node vector representation in the training set is used as the node feature, which is input into the binary logistic regression model for experiments. Finally, the prediction performance of the connected edge symbols on the prediction model of the test set is counted.

**Algorithm 1** The Overall Process of SNEMA

**Input:** signed network diagram  $G = (V, E)$ ,  
node feature  $\{h_i, V_i \in V\}$ ,  
node neighbor set  $\{\phi_0, \phi_1 \dots \phi_m\}$

**Output:** node vector final representation  $Z_i$

**1 procedure begins:**

**2 for each**  $V_i \in V$  **do :**

**3** Get the set of neighbors of the node  $\{\phi_0, \phi_1 \dots \phi_m\}$

**4 for each**  $\phi_i \in \{\phi_0, \phi_1 \dots \phi_m\}$  **do :**

**5 for each**  $V_i \in N_i^\phi$  **do:**

**6** Calculate the weight coefficient  $a_{ij}^\phi$  with formula (2)

**7** Calculate the node embedding  $h_i^\phi$  with formula (3)

**8 end for**

**9 end for**

**10** Get the node embedding  $\{h_i^{\phi_1} \dots h_i^{\phi_m}\}$  under each motif through node level attention:

**11** Calculate the importance of the weight coefficient motif with formula (5) (6)

**12** The final vector representation of the node:

$$Z_i = \sum_{i=1}^m B_{\phi_i} \cdot h_i^{\phi_m}$$

**13** The loss value is calculated by the formula (8)

**14** The gradient is calculated by backpropagation to update the model parameters

**15 end for**

**16 end procedure**

**A. DATASET INTRODUCTION**

In the experiment, four real social network data sets bitcoin alps, bitcoin OTC, Slashdot, and epinions are used. These data can be downloaded from Stanford's large network dataset<sup>1</sup> website. Most of the experimental research on signed networks is based on these four experimental sets. Each edge of these data sets has the meaning of positive edge and negative edge.

Bitcoin-alphas<sup>2</sup> and bitcoin-otc<sup>3</sup> are interpersonal networks that use bitcoin for transactions on the platforms of alpha and OTC. On the network platform, bitcoin users are anonymous. Maintain users' reputation records and prevents transactions with fraudulent and risky users, members of bitcoin alpha and OTC scores other members on a level of  $-10$  (complete distrust) to  $+10$  (complete trust). These score values greater than zero are regarded as positive links, and those at zero are regarded as negative links to form a signed network.

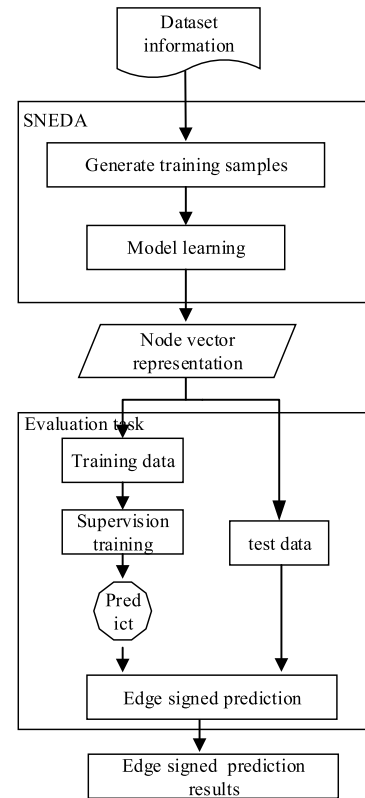
Slashdot<sup>4</sup> is a website of before and technology news. News can be provided by all users of the website. Users can choose their "friends" "enemies" here. They regard friends as positive relations and enemies as negative relations.

<sup>1</sup><http://snap.stanford.edu>

<sup>2</sup><http://snap.stanford.edu/data/soc-sign-bitcoin-alpha.html>

<sup>3</sup><http://snap.stanford.edu/data/soc-sign-bitcoin-otc.html>

<sup>4</sup><http://snap.stanford.edu/data/soc-sign-Slashdot090221.html>



**FIGURE 5.** Flow chart of SNEDA model experiment.

Epinions<sup>5</sup> is a consumer review website. Users can decide whether to trust another user by evaluating the quality of products, to users make better choices. All trust relationships and distrust relationships constitute this network, which regards the trust relationship as a positive relationship and the distrust relationship as a negative relationship.

**B. BASELINES**

The SNEMA model proposed in this paper is compared with the following benchmark methods. The following is an introduction to the benchmark methods:

- DeepWalk<sup>6</sup>: a network embedding method based on a random walk is designed for unsigned networks. Here, the negative links in the network are treated as positive links, and the whole signed network is regarded as an unsigned network.

- SiNE<sup>7</sup>: use the characteristics of the signed network to sample and model nodes with a random walk method to obtain node embedding.

- SiGAT<sup>8</sup>: provides a specific structure mode and uses gat to learn the nodes in each mode.

- SGCN<sup>9</sup>: use the structural balance theory to aggregate and spread information through graph convolution and generate node embedding.

<sup>5</sup><http://snap.stanford.edu/data/soc-sign-epinions.html>

<sup>6</sup><https://github.com/phanein/DeepWalk>

<sup>7</sup><http://www.public.asu.edu/%7Eswang187/codes/SiNE.zip>

<sup>8</sup><https://github.com/huangjunjie95/SiGAT>

<sup>9</sup><https://www.cse.msu.edu/%7Eederrtyle/code/SGCN.zip>

•SNEA<sup>10</sup>: learn the weight coefficients between nodes by using the self attention mechanism, and aggregate high-order information by using the balance theory to represent and learn the nodes.

•SDGNN<sup>11</sup>: Based on the structural balance theory in sociology, new information dissemination and aggregation mechanism is designed, and an average sampling theory is used to learn node embedding

### C. ASSESSMENT METRICS AND EXPERIMENTAL SETUP

In this paper, Accuracy, Macro-F1, F1, and AUC are used as evaluation metrics to validate the continuous edge sign prediction results. The higher values of these metrics indicate the more accurate prediction results of concatenated symbols.

**TABLE 2. Statistics of four datasets.**

Dataset	Nodes	pos links	neg links	%pos ration
Bitcoin-Alpha	3783	22490	1696	0.93
Bitcoin-Otc	5881	35592	3529	0.83
Slashdot	37626	313543	105529	0.74
Epinions	45003	513851	102180	0.85

The ratio of positive and negative relations and the number of neighbors in different social network datasets is not balanced, and the number of neighbors varies greatly among nodes in different datasets, as shown in Table3, which gives the optimal parameter settings in each of the four datasets.

**TABLE 3. Experimental parameter setting.**

	Bit-Alps	Bit-Otc	Slashdot	Epinions
Training rounds	500	500	1000	1000
Number of neighbors	100	200	500	500
node level dimension d	20	32	20	20
Path level dimension q	64	64	32	32
Learning rate lr	0.0008	0.0008	0.001	0.008
Weight_decay	0.0008	0.0008	0.001	0.001

### D. ANALYSIS OF EXPERIMENTAL RESULTS

The experimental comparison results of the SNEDA method and baseline method are shown in Table 4 and Table 5. From the chart, we can see that SNEDA is superior to the baseline method in all evaluation indicators on the four data sets, which indicates that the dual attention mechanism helps to improve the quality of node embedding. The following is a specific analysis:

a)The performance of the DeepWalk method, which ignores negative links for representation learning, is the worst, which shows that negative links affect the representation quality of signed networks to a great extent, and the representation method of unsigned networks is not suitable

for signed networks; SiNE based on sociological theory modeling is better than unsigned network, which shows the feasibility of modeling signed network based on structural balance theory.

b)SNEA introduces an attention mechanism to capture the different importance between nodes. The experimental results are better than SGCN (giving the same weight to node neighbors), which shows that assigning different weights to nodes through an attention mechanism can improve the quality of node embedding in signed networks.

c) SiGAT uses multi head attention to aggregate the information of first-order neighbors to obtain the embedding of target nodes, while SNEA aggregates the high-order neighbor information of nodes based on balance theory and uses the attention mechanism to learn the different weights between node pairs. From the experimental results, the SNEDA method is better in the network with fewer nodes, but not as good as SiGAT in the network with more nodes and is more complex. It may be because SiGAT only aggregates first-order neighbor information, while SNEA can aggregate higher-order neighbor information.

d)Compared with SiGAT, the SDGNN can capture high-order structure information with multiple layers. Experiments show that two-layer convolution can achieve the best effect; Both SDGNN and SNEA can process high-order structure information. SDGNN aggregates high-order structure information based on GraphSage, while SNEA aggregates high-order neighbor information through structure balance theory. From the experimental results, SDGNN is more effective than SNEA in aggregating high-order structure information.

e) Compared with SiGAT and SNEDA methods, which only focus on the attention weight between nodes, SNEDA adds path attention and considers the influence of different motifs. The experimental results show that the effect has been significantly improved. The results show that selectively aggregating neighbor information based on important relationship paths can improve the accuracy of link prediction, indicating that aggregating node neighbor information with dual attention can improve the quality of node embedding.

### E. ABLATION EXPERIMENTAL ANALYSIS

#### 1) ATTENTION MECHANISM

To verify the necessity of node-level attention and path-level attention, ablation experiments are carried out on the core components of the model in this section. The SNEDA-s method does not use path level attention, but directly uses node level attention to aggregate neighbor information. SNEDA uses node-level attention and path attention to obtain node embedding. The experimental results are shown in Table 6,the AUC equivalence of the SNEDA method has been significantly improved compared with SNEDA-S, which shows that the effect of node embedding is better when using a dual attention mechanism. The combination of node level attention and path attention can effectively improve the quality of node embedding and improve the accuracy of link prediction.

<sup>10</sup><https://github.com/liyu1990/SNEA>

<sup>11</sup><https://github.com/huangjunjie95/SDGNN>

TABLE 4. Experimental results.

Approaches	Bitcoin-Alpha				Bitcoin-OTC			
	Accuracy	F1	Macro-F1	AUC	Accuracy	F1	Macro-F1	AUC
DeepWalk	0.9265	0.9586	0.4957	0.6466	0.7368	0.8213	0.4184	0.5685
SiNE	0.9258	0.9689	0.6249	0.8628	0.8956	0.9204	0.6879	0.8571
SGCN	0.8106	0.8994	0.6635	0.8489	0.9016	0.9308	0.7031	0.8600
SNEA	0.9250	0.9712	0.6792	0.8800	0.8068	0.9488	0.7628	0.8749
SiGAT	<u>0.9455</u>	<u>0.9714</u>	0.680	0.8788	<b>0.9126</b>	<b>0.9568</b>	0.7158	0.8644
SDGNN	0.9395	0.968	0.6155	0.8566	0.8861	0.9357	0.7394	0.8507
SNEDA	<b>0.9546</b>	<b>0.9758</b>	<b>0.7376</b>	<b>0.9115</b>	0.9072	0.9518	<b>0.7489</b>	<b>0.8815</b>

TABLE 5. Experimental results.

Approaches	Slashdot				Epinions			
	Accuracy	F1	Macro-F1	AUC	Accuracy	F1	Macro-F1	AUC
DeepWalk	0.7118	0.8211	0.4256	0.5360	0.8149	0.8891	0.4556	0.5288
SiNE	0.7548	0.8579	0.6817	0.70991	0.8451	0.9156	0.5455	0.7328
SGCN	0.8075	0.8761	0.7325	0.8419	0.8734	0.9265	0.7316	0.8587
SNEA	0.8187	0.8818	0.7478	0.8609	0.892	0.937	0.783	0.8962
SiGAT	0.8273	0.8889	0.7507	0.8675	0.9035	0.9435	0.8074	0.9110
SDGNN	<u>0.8324</u>	<u>0.8925</u>	<u>0.7560</u>	<b>0.8737</b>	<u>0.9112</u>	<u>0.9475</u>	<u>0.8241</u>	<u>0.9214</u>
SNEDA	<b>0.8328</b>	<b>0.8926</b>	<b>0.7578</b>	<u>0.8710</u>	<b>0.9183</b>	<b>0.9520</b>	<b>0.8387</b>	<b>0.9306</b>

TABLE 6. Ablation experiment (attention).

Approachs	Bitcoin-Alphs			Bitcoin-OTC			Slashdot			Epinions		
	Ma-F1	Mi-F1	AUC	Ma-F1	Mi-F1	AUC	Ma-F1	Mi-F1	AUC	Ma-F1	Mi-F1	AUC
SNEDA-S	0.6906	0.9479	0.8933	0.6879	0.9311	0.8677	0.6680	0.9244	0.8432	0.8298	0.9138	0.9268
SNEDA	0.7676	0.9628	0.9142	0.7428	0.9566	0.8820	0.7266	0.9486	0.8720	0.8397	0.9187	0.9317

2) NODE NEIGHBOR ORDER

SNEDA-1 means to aggregate only the information of first-order neighbors, SNEDA-2 means to aggregate only the information of second-order neighbors, and SNEDA means to aggregate the information of first-order and second-order neighbors at the same time, as shown in Figure. 6. From the experimental results, it can be seen that aggregating the information of first-order and second-order neighbors at the same time can improve the link prediction results, which shows that aggregating the information of second-order neighbors can effectively improve the embedding quality of nodes.

F. SUPERPARAMETRIC ANALYSIS

SNEDA model has two important parameters to control the effect of the experiment, which are node embedding dimension d and path attention vector q. This section will analyze the influence of the selection of super parameters on the performance of SNEDA. Bitcoin alpha is selected as the experimental data set. The other parameter values are set to default values when analyzing specific super parameters.

Figure7 shows the AUC and F1 value of singled link prediction performance of the SNEDA model under different parameters. Figure7 (a) and 7 (b) show that with the increase of training rounds, the loss value gradually decreases, the AUC value increases, then gradually converges, and finally tends to be stable.

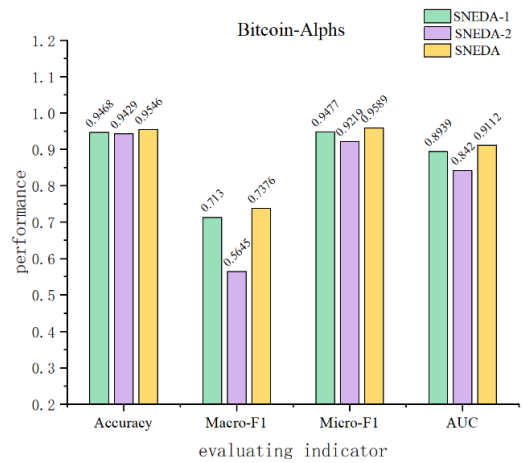


FIGURE 6. Neighbor order analysis.

Figure 8 shows the AUC value of singled link prediction performance of the SNEDA model under different parameters. Figure 8 (a) shows the influence of node attention vector d on the experimental results. It can be seen that the effect of vector dimension reaches the best at about 20. With the increase of dimension, the experimental effect decreases, which may be caused by overfitting Figure 8 (b) shows the impact of path vector dimension q on experimental performance. When the vector dimension is about 64, the effect reaches the best effect, and the effect decreases with the increase of vectthe or dimension.



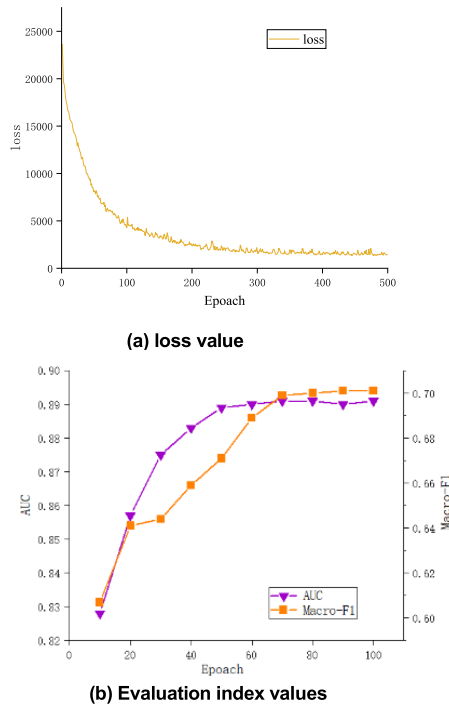


FIGURE 7. Hyperparametric analysis.

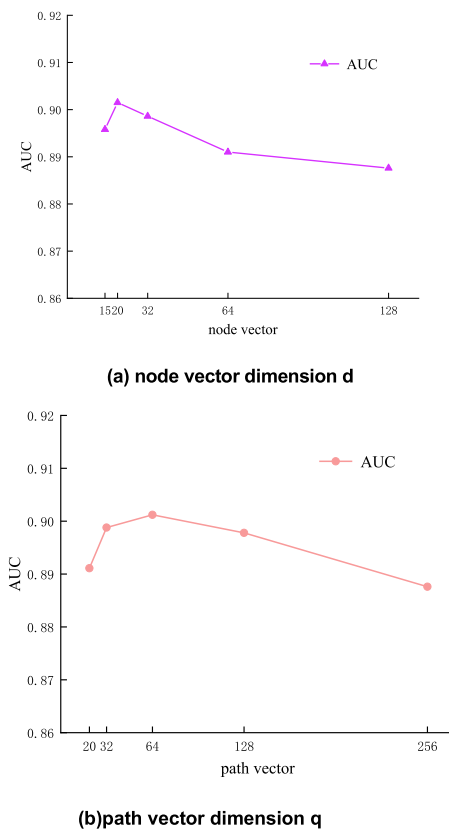


FIGURE 8. Hyperparametric analysis.

VI. SUMMARY

In this paper, a graph-based attentional network propagation SNEDA model is proposed to learn the node representation of the signed network. The model consists of two layers of

attention mechanism: node level attention learns node level information and learns different weights between nodes; Path level attention captures path level information and learns the importance of different motifs. Through the learning of the SNEDA model, the characteristic information of the target node contains both neighbor information and structure information, which makes the vector representation of the node more complete and rich. Compared with the baseline method, SNEDA achieves better results in the link prediction task, which shows that combining the path level information helps to improve the quality of node embedding, and proves the effectiveness of the SNEDA model. In future work, we will consider sampling higher-order neighbors, such as a random walk or GraphSage, to complete the integrity of node vectors.

REFERENCES

- [1] S. P. Borgatti, A. Mehra, D. J. Brass, and G. Labianca, "Network analysis in the social sciences," *Science*, vol. 323, no. 5916, pp. 892–895, 2009.
- [2] D. Camacho, A. Panizo-Lledot, G. Bello-Orgaz, A. Gonzalez-Pardo, and E. Cambria, "The four dimensions of social network analysis: An overview of research methods, applications, and software tools," *Inf. Fusion*, vol. 63, pp. 88–120, Nov. 2020.
- [3] D. Borsboom, M. K. Deserno, M. Rhemtulla, S. Epskamp, E. I. Fried, R. J. McNally, D. J. Robinaugh, M. Perugini, J. Dalege, G. Costantini, and A. M. Isvoranu, "Network analysis of multivariate data in psychological science," *Nature Rev. Methods Primers*, vol. 1, no. 1, pp. 1–18, 2021.
- [4] C. Liu, T. Arnon, C. Lazarus, C. Strong, C. Barrett, and M. J. Kochenderfer, "Algorithms for verifying deep neural networks," *Found. Trends Optim.*, vol. 4, nos. 3–4, pp. 244–404, 2021.
- [5] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1225–1234.
- [6] J. Zhou, L. Liu, W. Wei, and J. Fan, "Network representation learning: From preprocessing, feature extraction to node embedding," *ACM Comput. Surveys*, vol. 55, no. 2, pp. 1–35, Mar. 2023.
- [7] P. Cui, X. Wang, J. Pei, and W. Zhu, "A survey on network embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 5, pp. 833–852, May 2019.
- [8] J. Chen, M. Zhong, J. Li, D. Wang, T. Qian, and H. Tu, "Effective deep attributed network representation learning with topology adapted smoothing," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 5935–5946, Jul. 2022, doi: 10.1109/TCYB.2021.3064092.
- [9] T. Hiller, "Friends and enemies: A model of signed network formation," *Theor. Econ.*, vol. 12, no. 3, pp. 1057–1087, Sep. 2017.
- [10] Y. Zhang, P. Tino, A. Leonardis, and K. Tang, "A survey on neural network interpretability," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 5, pp. 726–742, Oct. 2021.
- [11] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, "A survey of signed network mining in social media," *ACM Comput. Surveys*, vol. 49, no. 3, pp. 1–37, Dec. 2016.
- [12] T. Derr, C. Aggarwal, and J. Tang, "Signed network modeling based on structural balance theory," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2018, pp. 557–566.
- [13] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Mar. 2020.
- [14] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, "Network embedding as matrix factorization: Unifying DeepWalk, LINE, PTE, and node2vec," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, Feb. 2018, pp. 459–467.
- [15] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 701–710.
- [16] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 855–864.
- [17] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: A comprehensive review," *Comput. Social Netw.*, vol. 6, no. 1, pp. 1–23, Dec. 2019.

- [18] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.
- [19] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
- [20] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.
- [21] S. Yuan, X. Wu, and Y. Xiang, "SNE: Signed network embedding," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*. Cham, Switzerland: Springer, 2017, pp. 183–195.
- [22] F. S. Rizzi and M. Granitzer, "Signed heterogeneous network embedding in social media," in *Proc. 35th Annu. ACM Symp. Appl. Comput.*, Mar. 2020, pp. 327–335.
- [23] T. Derr, Y. Ma, and J. Tang, "Signed graph convolutional networks," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 929–934.
- [24] Y. Li, Y. Tian, J. Zhang, and Y. Chang, "Learning signed network embedding via graph attention," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 4772–4779.
- [25] J. Huang, H. Shen, L. Hou, and X. Cheng, "Signed graph attention networks," in *Proc. Int. Conf. Artif. Neural Netw.* Munich, Germany: Springer, 2019, pp. 566–577.
- [26] J. Huang, H. Shen, L. Hou, and X. Cheng, "SDGNN: Learning node representation for signed directed networks," 2021, *arXiv:2101.02390*.
- [27] M. Brusco, P. Doreian, A. Mrvar, and D. Steinley, "Two algorithms for relaxed structural balance partitioning: Linking theory, models, and data to understand social network phenomena," *Sociol. Methods Res.*, vol. 40, no. 1, pp. 57–87, Feb. 2011.
- [28] P. Anchuri and M. Magdon-Ismail, "Communities and balance in signed networks: A spectral approach," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2012, pp. 235–242.
- [29] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 641–650.
- [30] W. Lin and B. Li, "Status-aware signed heterogeneous network embedding with graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Feb. 28, 2022, doi: [10.1109/TNNLS.2022.3151046](https://doi.org/10.1109/TNNLS.2022.3151046).
- [31] J. F. Stolte and R. M. Emerson, "Structural inequality: Position and power in network structures," in *Behavioral Theory in Sociology*. London, U.K.: Routledge, 2021, pp. 117–138.



**ZEKUN LU** (Member, IEEE) was born in 1996. He received the M.S. degree. He is currently pursuing the master's degree with North Minzu University. His research interests include social network analysis and network presentation learning.



**QIANCHENG YU** (Member, IEEE) was born in 1976. He received the Ph.D. degree. He is currently an Associate Professor and the M.S. Supervisor. His research interests include social perception computing, social network analysis, and machine learning.



**XIA LI** (Member, IEEE) was born in 1998. She is currently pursuing the master's degree with North Minzu University. Her research interests include machine learning and model detection.



**XIAONING LI** (Member, IEEE) was born in 1994. He received the M.S. degree. He is currently pursuing the master's degree with North Minzu University. His research interests include machine learning and data mining.



**AO QIANGWANG** (Member, IEEE) was born in 1999. He received the M.S. degree. He is currently pursuing the master's degree with North Minzu University. His research interests include machine learning and sequential database.

...