ABSTRACT

As an important tool of social network analysis, network representation learning also called network embedding maps the network to a latent space and learns low-dimensional and dense real vectors of nodes, while preserving the structure and internal attributes of network. The learned representations or embedding vectors can be used for node clustering, link prediction, network visualization and other tasks for network analysis. Most of the existing network representation learning algorithms mainly focus on the preservation of micro or macro network structure, ignoring the mesoscopic community structure information. Although a few network embedding methods are proposed to preserve the community structure, they all ignore the prior information about communities. Inspired by the semi-supervised community detection in complex networks, in this article, a novel Semi-Supervised DeepWalk method (SSDW) is proposed for network representation learning, which successfully preserves the community structure of network in the embedding space. Specifically, a semi-supervised random walk sampling method which effectively integrates the pairwise constraints is proposed. By doing so, the SSDW model can guide the transition probability in the random walk process and obtain the node context sequence in line with the prior knowledge. The experimental results on eight real networks show that comparing with the popular network embedding methods, the node representation vectors integrating pairwise constraints into the random walk process can obtain higher accuracy on node clustering task, and the results of link prediction, network visualization tasks indicate that the semi-supervised model SSDW is more discriminative than unsupervised ones.

INDEX TERMS

Network representation learning, semi-supervised, pairwise constraints, community structure, random walk.

I. INTRODUCTION

Complex network analysis is of great significance to study different systems, to explore the complex relationships among social networks, and to discover the attributes and contents of academic networks. With the exponential growth of internet data scale, the relationship between nodes in networks becomes more and more complex. However, the traditional network analysis technologies are difficult to adapt to large-scale data mining and machine learning tasks [1]. Network representation learning, also called network embedding, can preserve the structure and internal attributes of the network and learn effective low-dimensional representation vectors for nodes. The obtained node representation vectors can be used as node features to carry out subsequent network analysis tasks, such as community detection, node classification, link prediction, network visualization, and so on [2].

In recent years, a lot of methods about network representation learning are proposed. DeepWalk [3] is the most representative network embedding algorithm. It uses the random walk strategy to obtain node sequence of the network and then uses the famous Skip-gram model to train the node representation vectors. Line [4] describes the first-order and...
second-order similarity of nodes with two different functions from the perspective of network topology, and then combines them to get the final node representation vectors. Furthermore, GraRep [5], NEU [6], and AROPE [7] all capture the higher-order approximation except the first-order and the second-order similarity. But the network embedding methods mentioned above only preserve the local or global network structure.

Community structure is one of the most common and important topological properties of the network, at the same time, it is an mesoscopic description of the network structure. For nodes in the same community, even if they have no direct connection in topology structure, their representation vectors also should be similar. In recent years, a few network embedding methods which preserve the community structure are proposed. For example, MNMF [10] preserves the community structure of the network while considering the proximity of nodes. CARE [11] designs the community-aware random walks when learning the node representation vectors. Based on Expansion Sampling(XS) strategy [12]–[14], SENE [15] saves the mesoscopic community information in the learned node representation vectors.

However, the above network embedding methods which preserve community structure are unsupervised mode, they all ignore the prior information about community structures. There are two limitations in this unsupervised mode. Firstly, the learned node representation vectors have a great deviation in network analysis tasks, especially node clustering and classification. Secondly, most of the current network embedding algorithms which preserve community structure are based on the global enhancement pattern, there may be overfitting problems in the learning process of node representation vectors. After mapping nodes to low dimensional space, the spatial location of nodes is too gathered, so the nodes lose discrimination in subsequent network analysis.

To avoid the above two limitations, the semi-supervised methods can be considered to learn node representation vectors. Combined with the advantages of supervised learning and unsupervised learning, using a large number of unlabeled data and a small number of labeled data, integrating part of labeled data as prior information into the process of network representation learning can not only improve the discrimination of node embedding vectors but also avoid the overfitting problem of global enhancement pattern, better distinguish nodes in different communities.

In practical application, there are some prior information such as individual labels and pairwise constraints can be easily obtained in advance. These prior information can reflect community structure of the network. The use of individual labels must know the specific community number of a given network, it is more suitable for node classification tasks; but the use of pairwise constraints is relatively simple, researchers only need to know the community relationship of two nodes. If two nodes belong to the same community, their relationship is must-link. If two nodes belong to different communities, their relationship is cannot-link [17], so the pairwise constraints are more suitable for network clustering mode. In the past, some semi-supervised community detection methods [16]–[20] can find community structures with higher accuracy under the help of pairwise constraints. Inspired by the semi-supervised community detection in complex networks, the pairwise constraints perhaps can be used to guide random walk process and improve the discrimination of node representation vectors to a certain extent. The network representation learning depicts the similarity between nodes, the more similar nodes have closer distance in the embedding space; and the pairwise constraints can reflect the community relationship of nodes, the nodes in the same community are naturally more similar, when carry out random walk on the network topology, the nodes should move to other nodes in the same community with higher probability, so the use of pairwise constraints is very effective for network representation learning.

Therefore, this article proposes a novel Semi-Supervised DeepWalk method(SSDW) for network representation learning. Specifically, the pairwise constraints(must-link and cannot-link) are integrated into unsupervised random walk strategy to change the transfer probability of nodes. In this way, the community structure of the network can be preserved better. The integration of prior knowledge makes the learned node representations more discriminative. The difference between SSDW and the traditional DeepWalk method lies in that the nodes in the same community can be considered as the next walk node of the current node, while the nodes with direct link but belong to different communities can not be regarded as the next walk node of the current node. By doing so, random walk sequences containing community structure information can be generated, the node representation vectors learned by Skip-gram model will be more helpful for node clustering task.

To summarize, the contributions of this article are as follows:

- A novel Semi-Supervised DeepWalk method(SSDW) is proposed for network representation learning, which successfully saves the community structure information of the network in the learned node representation vectors.
- A semi-supervised random walk sampling method is designed, which successfully changes the node transfer probability in the traditional random walk process by using the pairwise constraints.
- The model SSDW is compared with unsupervised DeepWalk and several others representative network representation learning algorithms in the tasks of node clustering, link prediction and network visualization. The experimental results show that SSDW can obtain higher clustering accuracy and better visualization results.

II. RELATED WORK

Traditional network representation learning algorithms started from the perspective of preserving the local or global
topology of the network. DeepWalk [3] is the most representative network embedding algorithm. This model proves that the frequency of nodes distribution in the truncated random walk sequence is similar to the frequency of words distribution appearing in the text. Therefore, the Skip-gram model in the field of word vector representation is used to learn representations for nodes in the network. Line [4] is first proposed to optimize the objective function that preserves the first-order and the second-order proximity of nodes during node representation learning. The direct connection of nodes characterizes the first-order proximity, the co-neighbors of nodes without direct connection characterize the second order proximity. GraRep [5] captures different k-order proximity by defining different loss functions, and then merges the node representations learned by each function. NEU [6] first proves that the existing network representation learning methods include two steps of proximity matrix construction and dimensionality reduction. By constructing a k-order proximity matrix preserves higher-order node similarity and improves network embedding performance. ARPOE [7] preserves the proximity of any order based on the Singular Value Decomposition(SVD) framework to obtain the final node representation.

In recent years, many works have pointed out that the traditional network representation learning algorithms only preserve the micro local network structure or the macro global network structure, and ignore the mesoscopic community structure information. The combination of network representation learning and community structure modeling can impose higher-level constraints on node representations. MNMF [10] successfully saves the community structure in the generated node representations and improves the quality of node representation vectors, it preserves the first and second-order similarities of nodes with Non-negative Matrix Factorization(NMF), models community structure with modularity maximization. CARE [11] uses community attributes to capture more network structure information. It designs a community-aware random walk strategy and uses the Skip-gram model to learn the node representations. SENE [15] is a network embedding model based on community sampling. Inspired by DeepWalk and Expansion Sampling(XS), SENE modifies graph sampling strategy to save the community structure, and then uses Skip-gram model to learn the representation vector for each node.

The above network representation learning methods which preserve community structure can significantly improve the discrimination of node representation vectors. But they are all unsupervised patterns and ignore the prior information such as individual labels or pairwise constraints. We envisage whether the prior information can help to improve the performance of network representation learning algorithms. In the field of community detection, a lot of methods use pairwise constraints to improve the accuracy of community detection and design semi-supervised community detection algorithms. For example, NMF-LSE [16], SNMF-SS [17] and PSSNMF [18] are all based on the NMF model, they incorporate pairwise constraints to guide the clustering of nodes and effectively improve the accuracy of community detection. SS-masterl [19] incorporates pairwise constraints into the adjacency matrix of the original network and uses symmetric non-negative matrix factorization method for community detection. In addition, the SemiAttractor [20] is proposed to integrate the pairwise constraints into the distance dynamics model to improve the performance of community detection.

Therefore, inspired by semi-supervised community detection which uses pairwise constraints, the prior information–pairwise constraints can be integrated into unsupervised network representation learning methods, in this way, the community structure is well preserved when performing network representation learning.

In this article, a network representation learning method based on semi-supervised random walk is designed, the semi-supervised learning model not only improves the discrimination of node representation vectors learned by unsupervised embedding model, but also avoids the over-fitting problem of existing network embedding algorithms which save community structure. The semi-supervised random walk effectively uses pairwise constraints to guide the traditional random walk process, the sampled node context sequence contains the real community structure information of the network. Therefore, the node representation vectors learned by Skip-gram model are more discriminative in node clustering, link prediction and visualization tasks.

III. METHODOLOGY
A. NOTATIONS AND DEFINITIONS
In order to better describe the network embedding model based on semi-supervised random walk, the following definitions are first proposed:

Definition 1: (network embedding) Given an undirected and unweighted network \( G = (V, E) \), \( V \) represents the set of nodes and \( E \) represents the set of edges. The purpose of network embedding is to find a mapping function \( f : V \rightarrow U_{k} \subset \mathbb{R}^{d} \), finally learning the node representation matrix \( U_{k} \in U \) is the representation vector of corresponding node in low dimensional space, \( k \) is far less than the number of nodes \( |V| \).

Definition 2: (random walk) Most network embedding methods use random walk sequence to learn the representation vectors of nodes, and use Skip-gram model to train the random walk paths sampled from the network structure. A random walk sequence \( W_{v_{j}} \) with \( v_{i} \) as the root node is defined, a random walk sequence represents a path in a given network, so the random variable \( W_{v_{i}}^{1}, W_{v_{i}}^{2}, \ldots, W_{v_{i}}^{k} \) are used to represent the random walk sequence of node \( v_{i} \), and \( W_{v_{i}}^{k} \) represents the next walk node that randomly selected from the neighborhood of node \( v_{i} \).

Definition 3: (pairwise constraint matrix) For must-link constraint, matrix \( C_{ml} \) is defined, if \( C_{ml}(i, j) = 1 \), then node \( v_{i} \) and node \( v_{j} \) belong to the same community; similarly, for
cannot-link constraint, matrix $C_{cl}$ is defined, if $C_{cl}(i, j) = 1$, then node $v_i$ and node $v_j$ belong to different communities.

### B. FRAMEWORK

This article proposes a network representation learning model SSDW based on semi-supervised random walk. By using the pairwise constraints information related to the real community structure of network, node neighborhood structures considered in the traditional random walk model are changed, which affects the transfer probability of the next node in random walk process. Considering the community structure of network, a certain number of semi-supervised random walk sequences are generated, which guides the process of learning node eigenvectors in the traditional unsupervised network embedding model. The generated node representation vectors are more in line with prior knowledge, which improves the performance of network representation in node clustering task.

The pseudo code of SSDW can be shown as Algorithm 1.

**Algorithm 1** SSDW($G, w, k, \mu, l, C_{ml}, C_{cl}$)

Input:
- Graph $G = (V,E)$,
- Window length $w$,
- Representation size $k$,
- Number of random walks per node $\mu$,
- Pairwise Constraints Matrix $C_{ml}$ and $C_{cl}$.

Output:
- Matrix of node representations $\Phi \in R^{|V|\times k}$

1. sample $f$ from $U^{|V|\times k}$
2. while ($i < \mu$)
3.   $S = shuffle(V)$
4.   for each $v_i \in S$
5.     $W_{v_i} =$Semi-supervised RW($G, C_{ml}, C_{cl}, v_i, l$)
6.     Skip-gram($\Phi, W_{v_i}, w$)
7. end for

The learning flow chart of SSDW model can be shown as Figure 1.

As shown in Figure 1. Firstly, the semi-supervised SSDW model uses the original topology of the network and the prior information (must-link matrix and cannot-link matrix) as input; secondly, the SSDW model performs semi-supervised random walk process, and uses the pairwise constraints information which reflect the community structure to guide the traditional unsupervised random walk, and then obtains the semi-supervised random walk sequences, just like the second module in Figure 1. It is shown that the random walk sequences sampled by SSDW model can contain similar nodes belonging to the same community although there is no direct edge between them; thirdly, the SSDW model sends the sampled semi-supervised random walk sequences to the representative Skip-gram model for learning; and fourth, the SSDW model can obtain the node representation vectors more in line with the prior knowledge, and after embedding the nodes into two-dimensional space, the nodes belong to the same community can be embedded closer.

In order to make better use of prior information to guide the learning process of node representation, a semi-supervised network representation learning model is designed according to the following ideas:

1. As the most typical network representation learning model, DeepWalk first uses the random walk strategy to obtain the context sequence of nodes, but the random walk uses in this model is unsupervised. When selecting the next node, the neighbor nodes of current node are randomly selected with equal probability.

2. The random walk strategy used in DeepWalk does not consider the community structure information of the network. In practical application, some prior knowledge can be obtained easily such as pairwise constraints. Using prior information to guide the node transfer probability of random walk, semi-supervised random walk sequences can be obtained. The sampled node paths contain the real community structure.
(3) Taking the semi-supervised random walk sequences which preserve the community structure as the input of Skip-gram model, the embedding vectors of nodes can be well trained and more discriminative network representations can be obtained.

C. DeepWalk MODEL BASED ON UNSUPERVISED RANDOM WALK

DeepWalk algorithm proves theoretically that the frequency of nodes in random walk sequence is consistent with that of words in natural language, both of them obey power-law distribution. Therefore, when learning the representation vectors of nodes in the network, researchers can use the basic principle of word embedding model to simulate nodes in the network as words in the language model, and the sequence of nodes obtained by random walk can be simulated as sentences to be sent into the Skip-gram to get the final node representation vectors. The flow of DeepWalk algorithm mainly includes the following steps [3].

- Input the graph structure of the given network.
- Perform random walk on the graph structure. A fixed number of random walk sequences are generated for each node in the graph. The length of all walk sequences is $t$. Each step randomly selects the next walk node from the direct neighbors of the current node.
- Map the representation of nodes. For a random walk sequence, the Skip-gram model first maps the central node $v_i$ to a representation vector, and then defines a window of size $w$, if $w = 1$, then maximizes the probability $P_r(v_{i-1} | \Phi(v_i))$ and $P_r(v_{i+1} | \Phi(v_i))$.
- Hierarchical Softmax. In order to get the probability shown in Step 3, each update requires $O(v)$ operations, so the calculation of $P_r(\mu_k | \Phi(v_j))$ is very time-consuming. The Hierarchical Softmax function regards all nodes in the figure as leaf nodes of the balanced binary tree, then probability maximization of $P_r(v_{i-1} | \Phi(v_i))$ is equivalent to maximizing the probability of path from root node to current node.
- Get the final network representation. In the training process of this model, the node representation is initialized first, then the loss function is calculated, and finally the representation vectors of nodes and the weight parameters of the binary classification tree are learned.

D. SEMI-SUPERVISED RANDOM WALK SAMPLING METHOD

In the traditional unsupervised random walk strategy, it is assumed that $c_i$ represents the $i_{th}$ node in the walk sequence starting from the root node $c_0$, the transition probability matrix of random walk is as follows:

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{1}{|\text{neg}(v)|}, & \text{if } (v, x) \in E \\
0, & \text{otherwise} \end{cases}$$

where $|\text{neg}(v)|$ represents the number of $v'$ immediate neighbor nodes.

In undirected and unweighted graph, the traditional random walk models randomly select the neighbor nodes that have direct connection with the current node with equal probability. A simple network, as shown in Figure 2, is an example.

![Figure 2. A simple network which has three communities.](image)

If the current node of random walk sequence is node 1, because node 1 has three direct neighbors (node 2, 4, 5), then the probability of going to node 2, node 4, and node 5 in the next step is 1/3. However, in practical application, because node 1 and node 2 belong to different communities, the transition probability to node 2 should be reduced; in addition, node 6 and node 1 have no direct edge but they belong to the same community, so the transition probability to node 6 should be increased.

Different from unsupervised random walk strategy in DeepWalk, the pairwise constraints information can be effectively used to guide the random walk process. In order to obtain the semi-supervised random walk paths starting from node $c_0$, the immediate neighbors of each node are first extracted, and then the community structure information reflected by the pairwise constraint matrix $C_{ml}$ and $C_{cl}$ are integrated into the generation process of random walk sequences. If the relationship between node $v$ and $x$ belongs to must-link, node $v$ is added to the neighborhood of node $x$; if the relationship between node $v$ and $x$ belongs to cannot-link, node $v$ is removed from the neighborhood of node $x$. Specifically, the transition probability matrix of random walk is set to:

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{1}{|\text{neg}(v)\cup C_{ml}(v)\setminus C_{cl}(v)|}, & \text{if } (v, x) \in E \text{ and } (v, x) \in C_{ml} \text{ or } C_{cl} \\
0, & \text{otherwise} \end{cases}$$

where $C_{ml}(v)$ represents the set of $v'$ must-link nodes, $C_{cl}(v)$ represents the set of $v'$ cannot-link nodes.

The above operation successfully takes the real community structure information into account in the generation process of random walk sequences, which ensures that nodes belonging to the same community, though not connected, can be reached during random walk, and avoids sampling nodes belong to different communities with the current node. Specifically, the pseudo code of semi-supervised random walk strategy can be shown in Algorithm 2.
Algorithm 2 Semi-Supervised RW(G, C_ml, C_sl, c_0, l)
Input:
Graph G=(V,E),
Source node c_0 of RW,
Pairwise Constraints Matrix C_ml and C_sl,
Random walk max length l,
Output:
A path with max length l
1 initialize RW with c_0
2 extract v' immediate neighbors from G
3 for x ∈ V
4 if (v, x) ∈ C_ml
5 add x to v' immediate neighbors
6 else if (v, x) ∈ C_sl
7 remove x from v' immediate neighbors
8 else v' neighbors== v' immediate neighbors
end for
10 while length(path)< l
11 if current node has neighbors
12 select x at random form v' neighbors
13 else
14 back track in the path and select the last node
which has neighbors that are not in the path
15 end while

Different from the unsupervised random walk strategy in DeepWalk, the pairwise constraints information obtained in advance can be effectively used to guide the random walk process. In order to obtain the semi-supervised random walk path from node v_i, the direct neighbors of v_i are first extracted, and then the community structure information reflected by the pairwise constraints matrices C_ml and C_sl are integrated. If the relationship between v_i and v_j belongs to must-link, v_j will be added to the neighborhood structure of v_i; if the relationship between v_i and v_j belongs to cannot-link, v_j will be removed from the neighborhood structure of v_i.

Operations above successfully take the real community structure information into account in the generation process of random walk sequences, which ensure that nodes in same community can be reached during random walk, and avoid visiting the nodes that belong to different communities with the current node at the same time.

The pseudo code of semi-supervised random walk strategy is shown in Algorithm 2.

The Semi-supervised RW continues until the predefined path length l is met. If the neighborhood structure of the current node is empty, the expansion of path will be stopped.

Due to the use of pairwise constraints information related to community structure, the semi-supervised random walk paths preserve the local and global structure of the network at the same time. In addition, because the generated semi-supervised random walk paths are independent of each other, if some new nodes are added to the current network or some nodes are removed from network, new semi-supervised random walk sequences for previous nodes do not need to be generated.

E. THE SKIP-GRAM MODEL
According to the semi-supervised random walk strategy in Section 3.4, after sampling a certain number of random walk paths, the Skip-gram model is also used to learn the representation vectors of nodes. Skip-gram is a language model, which maximizes the co-occurrence probability of words in a sentence with a predefined window w [3]. If the path of node µ_k is defined as a node sequence (b_0, b_1, ..., b_log|v|) in a binary tree (b_0 is the root node, b_log|v| = µ_k), the basic formula of Skip-gram is as follows:

\[ P_r(\mu_k \mid \Phi(v_j)) = \max_{j} \prod_{l=1}^{log|V|} P_r(b_l \mid \Phi(v_j)) \]  \hspace{1cm} (1)

For each node in the network, semi-supervised walk paths can be generated iteratively. Predefined window w will slide on the build path. By using the Hierarchical Softmax function to approximate the probability distribution of formula(1), the following formula(2) can be get:

\[ P_r(b_l \mid \Phi(v_j)) = 1/(1 + e^{-\Phi(v_j) \cdot \Phi(b_l)}) \] \hspace{1cm} (2)

Finally, the stochastic gradient descent method is used to optimize the parameters, and back propagation(BP) model is used to estimate the derivative. The learning rate is set to 2.5% in the initial training. With the increase of nodes number in training process, the learning rate will gradually decrease.

The pseudo code of Skip-gram can be shown in Algorithm 3.

Algorithm 3 Skip-Gram(Φ, W_vj, w)
1 for each v_j ∈ W_v do
2 for each µ_k ∈ W_v[j − w : j + w] do
3 \[ J(\Phi) = -logP_r(\mu_k \mid \Phi(v_j)) \]
4 \[ \Phi = \Phi - \alpha * \frac{\partial J}{\partial \Phi} \]
5 end for
6 end for

All possible random walk assignments in window w are realized by Skip-gram iteratively. Specifically, Skip-gram maps each node v_j to a representation vector Φ(v_j) ∈ R^d, and after getting the representation of node v_j, the conditional probability of its neighbor nodes can be maximized in the walking path. Finally, the representation vectors of all nodes are learned.

IV. EXPERIMENTS AND RESULTS
A. DATASETS
To verify the effectiveness of the proposed method, on the following eight real network data sets are adopted to conduct experiments, including Polbooks, WebKB (Cornell, Texas, Washington, Wisconsin) [10], Email, Citeseer [26]
and Cora [21]. The details of these data sets can be shown in Table 1.

### B. BASELINE METHODS

The model SSDW is compared with DeepWalk based on unsupervised random walk and the following other representative network representation learning algorithms.

- **DeepWalk [3]:** Learning node representations using random walks and skip-gram model.
- **Line [4]:** Learning node representation by optimizing the objective function of the first-order and second-order proximity of nodes.
- **GraRep [5]:** Learning node representations in conjunction with global structure information.
- **node2vec [8]:** Learning the node representation by maximizing the possibility of saving the neighborhoods of nodes.
- **MNMF [10]:** Integrating node similarity and modularity-based community detection to learn node representation.
- **SSDW:** Incorporating pairwise constraints information to perform semi-supervised random walk. Through the community structure information reflected by pairwise constraints, the neighborhood structure of nodes is modified, and then the sampling probability of the next node is changed, which affects the final learned node representation vectors.

In order to make the comparison fair, the representation dimension \(d\) is set to 100 uniformly; the values of \(\alpha\) and \(\beta\) in MNMF model are set to 0.1 and 1 respectively. In order to verify the effect of different amounts of prior information on the representation of nodes, the ratio of prior information is set to 1\%, 2\%, 5\%, 10\%, named as SSDW(1), SSDW(2), SSDW(5) and SSDW(10) respectively. When the ratio of prior information is set to 0, SSDW is equivalent to the unsupervised model DeepWalk.

This article use different proportion of pairwise constraints to prove that the network representation learning model based on semi-supervised random walk can learn more discriminative node representation vectors than network representation learning model based on unsupervised random walk, and the more pairwise constraints information is used, the better node representation vectors perform in subsequent network analysis tasks.

### C. THE USAGE OF PRIOR INFORMATION

Given an undirected graph \(G\) having \(n\) nodes and \(k\) communities, there are \(N = \frac{n(n - 1)}{2}\) pairs of pairwise constraints, the constraints are divided into two types: must-link and cannot link. The total number of must-link is \(N_{ml} = \sum_{c=1}^{k} \frac{n_c(n_c - 1)}{2}\), where \(n_c\) represents the number of nodes included in the \(c\)-th community. The total number of cannot-link is \(N_{cl} = N - N_{ml}\). When we choose pairwise constraints, we randomly select two nodes from set \(V\). Different percentages 1\%, 2\%, 5\% and 10\% represent the random selection of a certain proportion of edge relationships from \(N = \frac{n(n - 1)}{2}\) pairwise constraints, and the \(C_{ml}\) and \(C_{cl}\) matrices of nodes are constructed according to the selected pairwise constraints.

### D. EXPERIMENT SETUP

#### 1) NODE CLUSTERING EXPERIMENT

In order to evaluate the effect of node representation, node clustering experiments is first used to verify the superiority of our model. First, the learned node representation vectors
are clustered by the k-means method, and then accuracy [22] is used to compare the node clustering results.

Accuracy is used to measure the clustering performance. Given a node \(v_i\), let \(r_i\) be the obtained cluster label, \(s_i\) be the label provided by the datasets that express community number which the node belongs, the accuracy is defined as follows:

\[
accuracy = \frac{\sum_{i=1}^{n} \delta(s_i, \text{map}(r_i))}{n}
\]

where \(n\) is the total number of network, \(\delta(x, y)\) is the delta function, that equals one if \(x = y\) and equals zero otherwise, and \(\text{map}(r_i)\) is the permutation mapping function that maps each cluster label \(r_i\) to the equivalent label from the community datasets. The best mapping can be found by using the Kuhn-Munkres algorithm [23].

In view of the sensitivity of k-means method to the initial values, the clustering experiments are repeated 20 times, each time with a new initial centroid, taking the average of 20 experimental results, as shown in Table 2.

It can be seen from the results in Table 2 that after using pairwise constraints to guide the random walk process, the accuracy of node embedding results obtained by Skip-gram model in node clustering task is better than that of unsupervised DeepWalk model, also better than Line, node2vec, MNMF and other baseline methods in Polbooks, Citeseer and Cora datasets, and with the increase of selection proportion of pairwise constraints, the clustering accuracy is higher and higher; especially in the largest Cora dataset, only 1% prior information is selected to guide the random walk process, compared with the unsupervised DeepWalk model, the improvement of clustering accuracy has reached nearly 65%. In four smaller data sets Cornell, Texas, Washington and Wisconsin, when only 1% of the prior information is selected, the clustering accuracy fails to improve, but after 2% of the prior information is selected, the improvement effect of clustering accuracy becomes obvious immediately. The specific effect of clustering accuracy increased with the increase of pairwise constraints selection proportion can be shown in Figure 2.

2) LINK PREDICTION EXPERIMENT

For the link prediction task, a node pair which is not connected as a negative instance for each edge is randomly sampled, while the links are considered positive instances. Then 70% of instances are randomly split as training set and the remaining instances as the test set. Then the node embeddings are learned on the training set and edge embeddings are generated by concatenating the two node embedding of links. Finally, the embeddings of edges are treated as features and whether or not a node pair has edges as the ground truth. A simple logistic regression classifier is trained on the training set and area under the ROC curve (AUC-ROC) which have been used in the previous work [24] is adopted to evaluate the performance on the test set. The results are shown in Table 3 compared with the baseline methods in the link prediction task.

From the results of link prediction task on eight datasets, the effect of SSDW model based on semi-supervised random walk on link prediction task is better than DeepWalk model based on unsupervised random walk. It is proved that using pairwise constraints information to guide random walk process can generate more discriminative node representation vectors. The reason why the performance of SSDW model are inferior to GraRep, MNMF, and other representative algorithms in link prediction task is that the use of...
pairwise constraints is more suitable for clustering mode, guiding nodes to move to the nodes within the same community as much as possible in the process of random walk, so that the nodes within each community gather more closely. When the link prediction task predicts whether there is a link between nodes, it is likely that the nodes within the same community but without a link are predicted to have a link, which results in the lower AUC-ROC value.

3) VISUALIZATION EXPERIMENT
In order to prove that the node representation vectors learned by semi-supervised SSDW model in this article is more interpretable, t-SNE visualization tool is used to show the effect of node embedding vectors learned by different network representation learning models on Polbooks dataset. In all graphs, each point represents a node in the network, and each color represents a category. The results shown in Figure 4 prove that using pairwise constraints to guide the random walk process can help Skip-gram model learn more discriminative node representation vectors. The SSDW model based on semi-supervised random walk can make the nodes of the same community gather more closely and the nodes of different communities distinguish more clearly. However, the node representations learned by other baseline methods cannot distinguish different types of nodes.

V. CONCLUSION
In this article, considering that the traditional network embedding model is unsupervised in the sampling process of random walk sequence, pairwise constraints (must-link and cannot-link) are used to guide the node transition probability in random walk process. With the prior information obtained in advance, if the next node and the current node belong to the same community, the sampling probability of next node should be increased; otherwise, the sampling probability of next node should be reduced. The semi-supervised random walk sequences are sent to Skip-gram model to learn the node representation vectors. The results of node embedding can obtain higher accuracy in node clustering task, and with the increase of pairwise constraints proportion, the clustering accuracy also improves. The results of the link prediction task also proves that the pairwise constraints information can effectively improve the discrimination of node representation vectors learned by DeepWalk model based on unsupervised random walk. The visualization results further show that the SSDW model based on semi-supervised random walk can make the nodes in the same community gather more closely. Our next work will focus on whether the pairwise constraints information can be well combined with other unsupervised node sequence sampling methods to improve the discrimination of node representation vectors.

REFERENCES
[1] 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.
[2] P. Goyal and E. Ferrara, “Graph embedding techniques, applications, and performance: A survey,” Knowl.-Based Syst., vol. 151, pp. 78–94, Jul. 2018.
[3] 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.
[4] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, “Line: Large-scale information network embedding,” in Proc. WWW, 2015, pp. 1067–1077.
[5] S. Cao, W. Lu, and Q. Xu, “GraRep: Learning graph representations with global structural information,” in Proc. 24th ACM Int. Conf. Inf. Knowl. Manage., 2015, pp. 891–900.

[6] C. Yang, M. Sun, Z. Liu, and C. Tu, “Fast network embedding enhancement via high order proximity approximation,” in Proc. Int. Joint Conf. Artif. Intell., 2017, pp. 3894–3900.

[7] Z. Zhang, P. Cui, X. Wang, J. Pei, X. Yao, and W. Zhu, “Arbitrary-order proximity preserved network embedding,” in Proc. KDD, 2018, pp. 2778–2786.

[8] 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.

[9] T. N. Kipf and M. Welling, “Graph convolutional networks,” 2016, arXiv:1609.02907. [Online]. Available: https://arxiv.org/abs/1609.02907

[10] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, “Community preserving network embedding,” in Proc. AAAI, 2017, pp. 3298239–3298270.

[11] M. M. Keikha, M. Rahgozar, and M. Asadpour, “Community aware random walk for network embedding,” Knowl.-Based Syst., vol. 148, pp. 47–54, May 2018.

[12] A. S. Maiya and T. Y. Berger-Wolf, “Benefits of bias: Towards better characterization of network sampling,” in Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), 2011, pp. 105–113.

[13] A. S. Maiya and T. Y. Berger-Wolf, “Expansion and search in networks,” in Proc. 19th ACM Int. Conf. Inf. Knowl. Manage. (CIKM), 2010, pp. 239–248.

[14] A. S. Maiya and T. Y. Berger-Wolf, “Sampling community structure,” in Proc. 19th Int. Conf. World Wide Web (WWW), 2010, pp. 701–710.

[15] Y. Yang, N. Ning, Y. Zhang, and B. Wu, “Community preserving node embedding based on seed-expansion sampling,” in Proc. IEEE 4th Int. Conf. Data Sci. Cyberspace (DSC), Jun. 2019, pp. 459–465.

[16] S. Zhang, R.-S. Wang, and X.-S. Zhang, “Uncovering fuzzy community structure in complex networks,” Phys. Rev. E. Stat. Phys. Plasmas Fluids. Relat. Interdiscip. Top., vol. 76, no. 4, Oct. 2007, Art. no. 046103.

[17] X. Ma, L. Gao, X. Yong, and L. Fu, “Semi-supervised clustering algorithm for community structure detection in complex networks,” Phys. A, Stat. Mech. Appl., vol. 389, no. 1, pp. 187–197, Jan. 2010.

[18] X. Liu, W. Wang, D. He, P. Jiao, D. Jin, and C. V. Cannistraci, “Semi-supervised community detection based on non-negative matrix factorization with node popularity,” Inf. Sci., vol. 381, pp. 304–321, Mar. 2017.

[19] Z.-Y. Zhang, K.-D. Sun, and S.-Q. Wang, “Enhanced community structure detection in complex networks with partial background information,” Sci. Rep., vol. 3, no. 1, Nov. 2013, Art. no. 3241.

[20] L. Fan, S. Xu, D. Liu, and Y. Ru, “Semi-supervised community structure detection based on distance dynamics,” IEEE Access, vol. 6, pp. 37261–37271, 2018.

[21] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore, “Automating the construction of Internet portals with machine learning,” Inf. Retr., vol. 3, no. 2, pp. 127–163, 2000.

[22] D. Cai, X. He, J. Han, and T. S. Huang, “Graph regularized nonnegative matrix factorization for data representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8, pp. 1548–1560, Aug. 2011.

[23] L. Lovasz and M. D. Plummer, Matching Theory (North-Holland Mathematics Studies), vol. 121. Budapest, Hungary: Akadémiai Kiadó, 1986.

[24] M. Gao, L. Chen, X. He, and A. Zhou, “BiNE: Bipartite network embedding,” in Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Jun. 2018, pp. 715–724.

[25] 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.

[26] C. Tu, “Max-margin DeepWalk: Discriminative learning of network representation,” in Proc. Int. Joint Conf. Artif. Intell., 2016, pp. 3889–3895.

[27] Z. Yang, W. Cohen, and R. Salakhutdinov, “Revisiting semi-supervised learning with graph embedding,” in Proc. ICML, 2016, pp. 40–48.

[28] M. E. J. Newman, “The structure and function of complex network,” SIAM Rev., vol. 45, no. 2, pp. 167–256, 2003.