Community Preserving Sign Prediction for Weak Ties of Complex Networks

Kangya He[1], Donghai Guan[1,2], and Weiwei Yuan[1,2](✉)

[1] College of Computer Science and Technology,
Nanjing University of Aeronautics and Astronautics, Nanjing, China
kangyahe@gmail.com, {dhguan,yuanweiwei}@nuaa.edu.cn
[2] Collaborative Innovation Center of Novel Software Technology and Industrialization,
Nanjing, China

Abstract. The weak ties are crucial bridges between the tightly coupled node groups in complex networks. Despite of their importance, no existing work has focused on the sign prediction of weak ties. A community preserving sign prediction model is therefore proposed to predict the sign of the weak ties. Nodes are firstly divided into different communities. The weak ties are then detected via the connections of the divided communities. SVM classifier is finally trained and used to predict the sign of weak ties. Experiments held on the real world dataset verify the high prediction performances of our proposed method for weak ties of complex networks.

Keywords: Sign prediction · Weak tie · Link prediction · Signed network

1 Introduction

One basic topology of the complex network is its small-worldness [1], i.e., nodes of the complex network could connect to each other within limited number hops of propagations. However, nodes usually have closed relationship with very limited number of other nodes. Nodes of the complex network cannot be widely connected to most nodes without the existence of weak ties. Weak ties are the links which connect different groups of users who have strong relationships. And the links connect nodes inside the groups in which users have strong relationships are called strong ties. The weak tie is not merely a trivial acquaintance tie between nodes, but rather a crucial bridge between the two densely knitted clumps of close friends [2].

Despite of the importance of weak ties, to the best of our knowledge, no existing work has focused on the sign prediction problem of weak ties in complex networks. Existing works of sign prediction predicts the signs of link in the complex network. A positive sign means the source node of the link trusts or likes the target node of the link. A negative sign means the source node of the link distrusts or dislikes the target node of the link. A common sign prediction method is to extract a set of attributes related to the links, train a classifier to learn the attributes and the related signs, and then predict the sign of the target link with given attributes according to the trained sign classifier. However, the target link of the sign prediction does not differentiate the weak ties and

L. Wang et al. (Eds.): QShine 2017, LNICST 234, pp. 13–22, 2018.
https://doi.org/10.1007/978-3-319-78078-8_2

the strong ties. Since the weak tie is curial for the connection of complex networks, this work focuses on the sign prediction of weak ties.

To predict the sign of the weak ties in complex networks, a community preserving sign prediction model is proposed in this work. Nodes are firstly divided into different communities. This is achieved by learning the weight of nodes, the belonging degree of nodes and the modularity of the complex network. The weak ties are then detected via the connections of the divided communities. To predict the sign of the detected weak ties, five attributes are extracted for each weak tie, including the Jaccard similarity, the negative outdegree ratio of the source node, the negative indegree ratio of the target node, the positive link ratio between communities, and the negative link ratio between communities. SVM classifier is finally trained and used to predict the sign of weak ties based on the extracted attributes. Experiments held on real world application dataset show that the proposed method has high sign prediction accuracy and high negative sign prediction F1-score for weak ties of complex networks.

The following of this paper is organized as follows: Sect. 2 introduces the related works, Sect. 3 gives the proposed method, Sect. 4 presents the experimental results and Sect. 5 concludes this paper and points out the future directions.

2 Related Works

The related works of sign prediction can mainly be divided into two categories. One uses the triad information of nodes in signed networks [3]. The other calculates the similarities between node and trains machine learning algorithms to predict the signs. The latter category of related works is more related to this work. Some of the most popular node similarity measurements are summarized as follows:

A. CN

CN [4] measures the similarity of users by the number of their common neighbors. The more common neighbors two nodes have, the more similar they are. Suppose node v_i and node v_j are two nodes of graph G, the CN similarity of v_i and v_j is:

$$S^{CN} = \left| N_1^G(v_i) \cap N_1^G(v_j) \right| \tag{1}$$

where $N_1^G(v_i)$ is the neighbors of v_i in G, $N_1^G(v_j)$ is the neighbors of v_j in G, and $|\bullet|$ means the number of $\bullet$.

B. RA

RA [5] is based on the idea of resource allocation. As mentioned in [6], the resource of each node is regarded as a unit; each node allocates its resource evenly to its neighbors, and the resource between each pair of nodes are transferred via their common neighbors. The similarity of two nodes are defined as the resource one node can get from the other node.

For node v_i and node v_j of graph G, their RA similarity is calculated as:

$$S^{RA} = \sum_{z \in N_1^G(v_i) \cap N_1^G(v_j)} \frac{1}{d(z)} \tag{2}$$

where $N_1^G(v_i)$ and $N_1^G(v_j)$ are the neighbors of v_i and v_j in G respectively, $d(z)$ is degree of the selected common neighbor.

The difference of CN and RA is that: CA does not differentiate the common neighbors, i.e., each common neighbor is supposed to have the same contribution to the similarity calculation; while RA differentiates common neighbors by their degrees. i.e., the higher degree a common neighbor has, the less important of this selected common neighbor is. This is because the higher degree a common neighbor has, the less resource it can allocate to the target node. RA sets the importance of the common neighbor linearly relate to the reciprocal of the common neighbor's degree.

C. AA

AA [7] is similar as RA: they both differentiates common neighbors by their degrees. The difference is that AA uses the logarithm of degrees to differentiate the contribution of common neighbors to user similarity, while RA directly uses the degrees to differentiate the contribution of common neighbors to user similarity. In some networks, the degrees of nodes tend to be very high, if the user similarity calculation uses the reciprocal of degrees directly, some similarity tends to be very small. AA therefore improves RA by enlarging the value of similarity. For node v_i and node v_j of G, their AA similarity is calculated as:

$$S^{AA} = \sum_{z \in N_1^G(v_i) \cap N_1^G(v_j)} \frac{1}{\log(d(z))} \tag{3}$$

3 The Proposed Sign Prediction Method for Weak Ties

The architecture of the proposed method is given in Fig. 1. The input is the graph representation of the complex network, and the outputs are the predicted signs for the weak ties. It consists of three modules. The details of these modules are given in the following subsections.

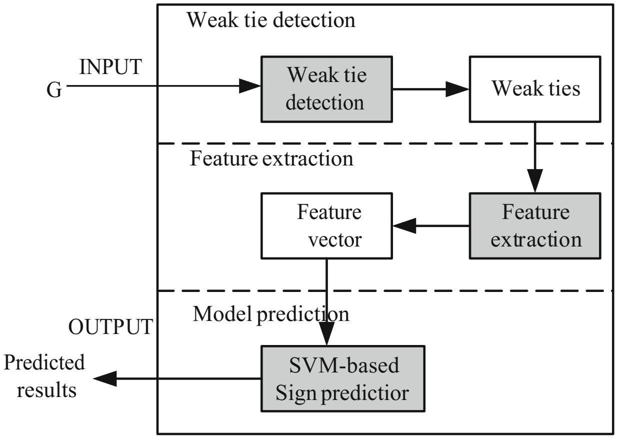

Fig. 1. The architecture of the proposed method

3.1 Weak Tie Detection

The weak tie detection module is based on the community detection method proposed by [8]. The algorithm is mainly based on the following attributes: A. The weight of nodes. It is described by the degree of the node in this work:

$$w(v) = \sum_{u \in \Gamma_{in}(v)} d_{in}(u) \tag{4}$$

where $\Gamma_{in(v)}$ is the set of nodes which have indegree in the network, and is the indegree of the node. B. The belonging degree of nodes. It represents the relationship between nodes and communities. If a node has high belonging degree with a community, this node will be categorized into this community:

$$B(i, C) = \frac{\sum_{j \in C} \left(w_{ij} + w_{ji}\right)}{\sum_{j \notin C} \left(w_{ij} + w_{ji}\right)} \tag{5}$$

where w_{ij} is the weight represents the connection from node i to node j: $w_{ij} = 1$ if there exists a directed edge from to, otherwise, $w_{ij} = 0$.

C. The modularity of the network. It is also known as the Q value. The bigger it is, the better performance of the community clustering:

$$Q = \frac{1}{m} \sum_{1 \le i,j \le n} \left[a_{ij} - \frac{k_i^{in} k_j^{out}}{m} \right] \delta\left(C_i, C_j\right) \tag{6}$$

where a_{ij} represents the existence of the edge pointing from node i to node j, $a_{ij} = 1$ if there exists a directed edge pointing from i to j, otherwise, $a_{ij} = 0$; m is the scale of E;

C_i and C_j represent the community of i and j respectively; $\delta(C_i, C_j)$ represents the consistency of C_i and C_j, $\delta(C_i, C_j) = 1$ if $C_i = C_j$, otherwise, $\delta(C_i, C_j) = 0$.

Using the above attributes, the algorithm given in Algorithm 1 is used to divide the communities of the complex network: the node with the largest weight, which is calculated by (4), is used as the initial community; the neighbors of the initial community are firstly involved in the community. For each neighbor of the updated community, calculating its belonging degree to this community, if it is bigger than 1, this neighbor is added to the community. This procedure is repeated until each node is involved in some community. The community division is then optimized by maximizing the modularity of the network, which is calculated by (6).

Algorithm 1 The algorithm of community division.

```
Input: $G = (V, E)$,$V$ and $E$ are the vertex set and edge set of the graph $G$
Parameters: $\Gamma(v)$ is the neighbor set of vertex $v$;$W$ is the array of weight, $n$ is the
    scale of $V$;$V_0$ is the nodes whose indegree is 0, $tC$ are communities.
Output: the community set $CS$.

$V \leftarrow V - V_0$
$CS \leftarrow \Phi$
while $V \neq \Phi$ do
    $C \leftarrow \Phi$
    Get the vertex $v_{max}$ with the largest weight $W(v)$ in $V$
    $C \leftarrow C \cup \{v_{\max}\} \cup \Gamma(v_{\max})$
    $V \leftarrow V - C$
    $tC \leftarrow \Phi$
    for $\forall v \in V$ do
        if $B(v, C) > 1$ then
            $tC \leftarrow tC \cup \{v\}$
        end if
    end for
    $C \leftarrow C \cup tC$
    $CS \leftarrow CS \cup \{C\}$
end while
for $\forall v_0 \in V_0$ do
    Get$G_i$ which has the most links with $v_0$
    $C_i \leftarrow C_i \cup \{v_0\}$
end for
Caculate $Q_1$ of $G$ with $CS$
$Q_2 \triangleq Q_1 + 1$
while $Q_1 < Q_2$ do
    Traversal $CS$ to find the largest $Q_2$ when merging $C_i$, $C_j$
    $t = C_i \cup C_j$
    $CS \leftarrow CS - \{C_i, C_j\}$
    $CS \leftarrow CS \cup \{t\}$
end while
Return $CS$
```

With the communities divided by the algorithm shown in Algorithm 1, the weak ties of the network are detected, as shown in Algorithm 2. For the signed directed network $G = (V, E)$, a positive signed network $G^+ = (V_1, E_1)$ and a negative signed network

$G^- = (V_2, E_2)$ are first extracted for the weak tie detection. G^+ is composed by all the positive edges of E, and G^- is composed by all the negative edges of E. Using the algorithm given in Algorithm 1, two sets of communities CS^+ and CS^- are divided, in which CS^+ is the communities divided by G^+ and CS^- is the communities divided by G^-. The weak tie detection algorithm traverses each edges of E, if two end nodes of the edge belong to one community, this edge is regarded as the strong tie; otherwise, if two end nodes of the edge belong to two communities, this edge is regarded as the weak tie.

Algorithm 2 The algorithm of weak ties detection.

```
Input: $G = (V, E)$,$V$ and $E$ are the vertex set and edge set of the graph $G$
Parameters: $CS^+$, $CS^-$ the positive and negative community set.
Output: $U$ the community set.

$U \leftarrow \Phi$
$E = \{e_1, e_2, \cdots, e_m\}$
$m = |E|$
for $i = 1$ to $m$ do
    $(s, t) \leftarrow e_i$
    if the vertec $s$,$t$ belong to different communities in $CS^+$, $CS^-$ then
        $U \leftarrow U \cup e_i$
    end if
end for
Return $U$
```

3.2 Feature Extraction

For each weak tie extracted by the above section, several attributes are extracted for the further sign prediction:

A. Jaccard similarity. The more similar two nodes are, the more possible the sign of the link connecting these two nodes is positive. The less similar two nodes are, the more possible the sign of the link connecting these two nodes is negative. It is calculated as:

$$JC\left(v_i, v_j\right) = \frac{|\Gamma(v_i) \cap \Gamma(v_j)|}{|\Gamma(v_i) \cup \Gamma(v_j)|} \tag{7}$$

where $\Gamma(\bullet)$ is the set of neighbors of • and $|\bullet|$ is the number of nodes in.

B. Negative outdegree ratio of the source node. The higher it is, the more possible the sign of the weak tie is negative. It is calculated as:

$$NOR(s) = \frac{d_{out}^{-}(s)}{d_{out}^{-}(s) + d_{out}^{+}(s)} \tag{8}$$

where s represents the source node of the weak tie, $d_{out}^{-}(s)$ is the negative out-degree of s, and $d_{out}^{+}(s)$ is the outdegree of s.

C. Negative indegree ratio of the target node. The higher it is, the more possible the sign of the weak tie is negative. It is calculated as:

$$NIR(t) = \frac{d_{in}^{-}(t)}{d_{in}^{-}(t) + d_{in}^{+}(t)} \tag{9}$$

where t represents the target node of the weak tie.

D. Positive link ratio between communities. The higher it is, the more likely the target weak tie is positive. It is measured as:

$$R^{+}\left(C_i, C_j\right) = \frac{P\left(C_i, C_j\right)}{N\left(C_i, C_j\right) + P\left(C_i, C_j\right)} \tag{10}$$

where P (C_i, C_j) is the number of positive links between community C_i and community C_j, and N (C_i, C_j) is the number of negative links between C_i and C_j.

E. Negative link ratio between communities. The higher it is, the more likely the target weak tie is negative. It is measured as:

$$R^{-}\left(C_i, C_j\right) = \frac{N\left(C_i, C_j\right)}{N\left(C_i, C_j\right) + P\left(C_i, C_j\right)} \tag{11}$$

3.3 Sign Predictor

Using the features extracted for each target weak tie, the SVM classifier is applied to predict the sign of the weak tie. Based on the featured extracted as shown in Sect. 3.2, a vector is generated for each target weak tie, i.e. $\mathbf{x} \in \mathbb{R}^5$ is used to describe the weak tie. Let be the sign of the weak tie, $y \in \{+1, -1\}$, in which $+1$ means the sign of the weak tie is positive, and -1 means the sign of the weak tie is negative. $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_m, y_m)\}$ is used to train the classifier, in which $\mathbf{x}_i$ is the vector describing the i^{th} training weak tie, $\mathbf{x}_i \in \mathbb{R}^5, i = 1, 2, \cdots, m$, m is the number of weak ties used for the training of SVM classifier, and y_i is the sign of the i^{th} training weak tie. The sign of the weak tie is predicted as:

$$sign(f(\mathbf{x})) = sign(\omega^{\mathrm{T}}\mathbf{x} + b) \tag{12}$$

where $sign(\bullet)$ is the sign of $\bullet$, ω and b are weight of the attributes and the bias respectively.

4 Experimental Results

The performances of the proposed method are measured on the real world application data Epinions dataset [9]. Epinions is an online review website where users can not only give their ratings on items but also point out their opinions to other users. If a user trusts another user, the sign of the link connecting these two users is regards as positive. If a user distrusts another user, the sign of the link connecting these two users is regards as negative. The Epinions dataset consists of 131828 nodes and 841372 directed links between these nodes, in which 85.3% links have positive signs and 14.7% links have negative signs.

Since the original dataset is sparse, the data are firstly preprocessed for better sign prediction. The data preprocessing keeps the nodes whose degree are bigger than 80, and removes the nodes whose degree are less than 80, as well as the connections pointing to these nodes. The remaining experimental dataset contains 1415 nodes and 113484 links, in which 99376 links are positive and 14108 links are negative. A positive network and a negative network are extracted from this experimental dataset for further sign prediction, in which the positive network contains all positive links of the experimental dataset and the negative network contains all negative links of the experimental dataset. The positive network consists of 1414 nodes and 99376 links, and the negative network consists of 1346 nodes and 14108 links.

Using the method given in Sect. 3.1, the weak ties of the experimental dataset are firstly detected. Based on the algorithm given in Algorithm 1, totally 16 communities are divided for the positive network and 31 communities are divided for the negative network. Based on the weak tie detection algorithm given in Algorithm 2, 1882 weak ties are detected between these communities, in which 1275 weak ties have positive signs and 607 weak ties have negative signs. We randomly select 70% of the weak ties to train the sign classifier and the remaining 30% of the weak ties are used to test the performances of the proposed method. The experiments are repeatedly held on the

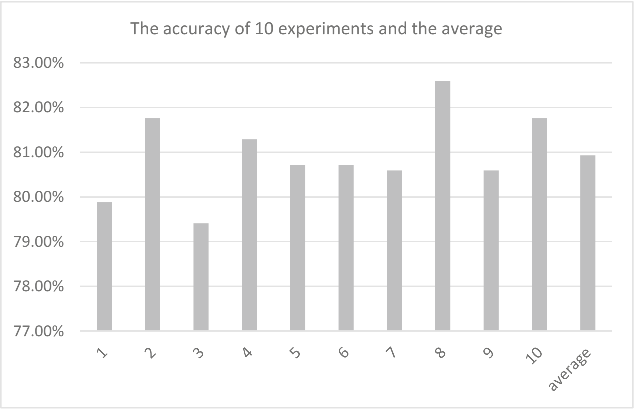

Fig. 2. The accuracy of the weak tie sign prediction

experimental data for ten times. The accuracy and the F1-score of the weak tie sign prediction are given in Figs. 2 and 3 respectively.

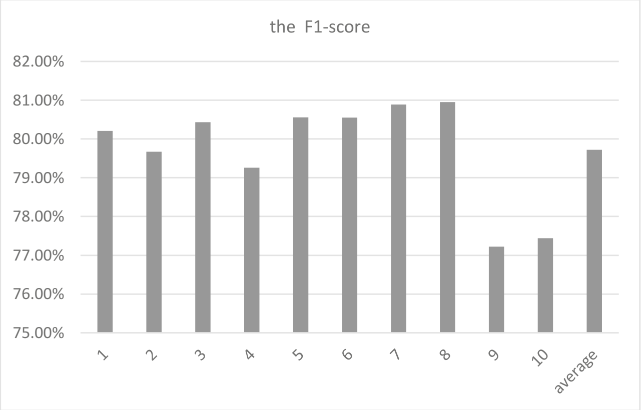

Fig. 3. The F1-Score of the negative sign prediction of the weak ties.

As shown in Fig. 2, the proposed method has high weak tie sign prediction accuracy. The average prediction accuracy of the 10 times experiments is 80.93%. For all experiments, the prediction accuracy is over 79%, and the prediction accuracy is more than 80% in 8 out of the 10 experiments. Since negative signs usually contain more information than positive signs [10], the performance of negative sign prediction is extremely important. We therefore measure the F1-score of the negative sign prediction for the weak ties, as shown in Fig. 3. The F1-score of the 10 times experiments is 79.72%. For all experiments, the F1-score is over 77%, and the F1-score is more than 79% in 8 out of the 10 experiments. Note that there is no existing work predicting the weak ties of signed network, so the performances of the proposed method could not be compared with the performances of other works.

5 Conclusions and Future Works

The sign prediction for weak ties of complex networks is a newly emerged research problem in the area of sign prediction. Weak ties are crucial bridges connecting tightly coupled nodes groups. The sign of the weak ties represents the relationships between two groups, which carries more information than the sign of strong ties in the complex networks. The paper propose a communited based sign predicting method to predict the sign of weak ties in complex network. The weak ties are firstly detected by community division. SVM classifier is trained to learn the relationship between the sign of the weak ties and the attributes related to the weak ties. The trained classifier is then used for the sign prediction of the unknown weak ties. Experimental results verifies the effectiveness of the proposed method in the real application data. Our future work will mainly focus on the performance improvement on the sign prediction of the weak ties. We will not only try to further improve the sign prediction accuracy, but also try to improve the F1-score of the negative sign prediction.

Acknowledgement. This research was supported by Nature Science Foundation of China (Grant No. 61672284), Natural Science Foundation of Jiangsu Province (Grant No. BK20171418), China Postdoctoral Science Foundation (Grant No. 2016M591841). This work was also supported by Open Project Foundation of Information Technology Research Base of Civil Aviation Administration of China (Grant No. CAAC-ITRB-201501 and Grant No. CAAC-ITRB-201602). Dr. Weiwei Yuan is the corresponding author of this paper.

References

1. Yuan, W., Guan, D., Lee, Y.K., et al.: Improved trust-aware recommender system using small-worldness of trust networks. J. Knowl. Based Syst. **23**(3), 232–238 (1981)
2. Wei, L., Xu, H., Wang, Z., et al.: Topic detection based on weak tie analysis: a case study of LIS research. J. Data Inf. Sci. **1**(4), 81–101 (2016)
3. Li, X., Fang, H., Zhang, J.: Rethinking the link prediction problem in signed social networks. In: AAAI, pp. 4955–4956 (2017)
4. Tang, J., Chang, Y., Aggarwal, C., et al.: A survey of signed network mining in social media. J. ACM Comput. Surv. (CSUR) **49**(3), 42 (2016)
5. Si, C., Jiao, L., Wu, J., et al.: A group evolving-based framework with perturbations for link prediction. J. Physica A Stat. Mech. Appl. **475**, 117–128 (2017)
6. Martnez, V., Berzal, F., Cubero, J.C.: A survey of link prediction in complex networks. J. ACM Comput. Surv. (CSUR) **49**(4), 69 (2016)
7. Nocaj, A., Ortmann, M., Brandes, U.: Adaptive disentanglement based on local clustering in small-world network visualization. J. IEEE Trans. Vis. Comput. Graph. **22**(6), 1662–1671 (2016)
8. Leicht, E.A., Newman, M.E.J.: Community structure in directed networks. J. Phys. Rev. Lett. **100**(11), 118703 (2008)
9. Massa, P., Avesani, P.: Trust-aware recommender systems. In: ACM Conference on Recommender Systems, pp. 17–24. ACM (2007)
10. Khodadadi, A., Jalili, M.: Sign prediction in social networks based on tendency rate of equivalent micro-structures. J. Neurocomput. **257**, 175–184 (2017)