

Chapter 44

Balance Theory-Based Model for Discovering Trust Network

Guangming Yang, Xiangjun Hou, Zhenhua Tan, Liangyu Zhang
and Hainan Yu

Abstract Relations among users on social network often reflect a mixture of positive (trust) and negative (distrust) interactions. It is necessary to extract the trust relationships among people and discover a trust network from the complex social network. The trust network can be used to calculate the trust confidence and can infer the trust values among the nodes in it. In this paper, we propose a method to estimate a trust network and it is the first time that the balance theory is applied to solve the problem of discovering trust network. The method makes use of balance theory to estimate the credibility of the relationships among the nodes. First, we analyzed the core content of the balance theory and the similarities with trust network. Then, we proposed the rules of building the trust network, and summarized the process of the implementation. The experiment runs on Epinions data set which consists of more than 130,000 nodes. The experiment results demonstrate that our algorithm performs well in building trust network.

Keywords Trust network · Balance theory · Trust · Distrust

44.1 Introduction

A social network is a set of nodes connected by a set of relationship, such as friendship, affiliation, or information exchange [1, 2]. People in the social network are transferring information every day; they do a lot of interactions. It becomes particularly important to extract the trust relationships among people in the social

G. Yang · X. Hou (✉) · Z. Tan · L. Zhang · H. Yu
Software College, Northeastern University, Shenyang 110819, China
e-mail: tangmeisuan@163.com

network. Social networks underlying current social media sites often reflect a mixture of positive and negative links. It means each link is created by one user to another, so the networks can be viewed as directed graphs. The trust network can be used to calculate the trust confidence and can infer the trust values between the nodes in it.

The experts have proposed many ways to build trust networks and solve the problem of trust computing. The CONFIDANT protocol [3, 4] is used to calculate the local historical information. Structured by a bidirectional circle topology, the p2p network has a short routing table to record one super node, one successor node, one previous node, and several cache nodes, but it cannot exclude the interference of the spy node [5]. The SUNNY algorithm that uses probabilistic sampling to separately estimates trust information and our confidence in the trust estimate and uses the two values in order to compute an estimate of trust based on only those information sources with the highest confidence estimates [6]. Although it produced more accurate trust estimates than the well-known trust inference algorithm TIDAL TRUST, it still has a lot of problems about the complex network path. Jøsang uses subjective logic to compute trust among arbitrary parties in the network, and proposed an analysis method of trust chains which simplifies the structure of the network to calculate according to the series-parallel networks [7, 8]. However, this method loses lots of trust information and costs too much communication overhead. In addition, there is a paper which describes a context-based trust model [9]. The model computes direct trust basing on the truster's direct or similar recommendation about a trustee with respect to a given context, but it does not compute recommended trust. While we propose to process the nodes first in order to both filter out redundant and useless nodes and to extract nodes with certain requirements from the original network.

In this paper, the main focus of our work here is to examine the interplay between positive and negative links in social network and describe a new method which is based on Hyde's balance theory to build a trust network. We analyzed the basic knowledge of the balance theory. Our algorithm is the first algorithm that applies the balance theory to discover a trust network. We conduct our experiment results on the Epinions data set, the results show that using balance theory to build a trust network performs very well.

44.2 Related Work

The proposed algorithm based on the balance theory in which it views all the relationships as the three edges of a triangle. Next, we define some symbols and notations that will be used to explain the algorithm.

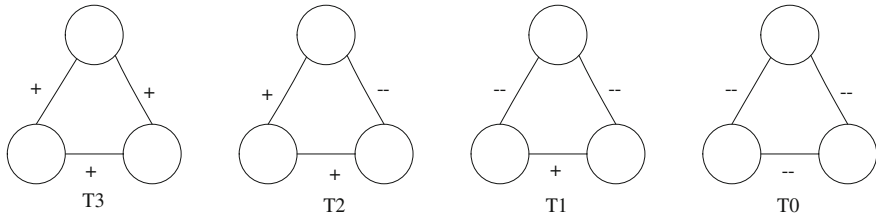


Fig. 44.1 There are four types of triples whose edges are with positive or negative signs

44.2.1 The Balance Theory

The balance theory which originated in social psychology in the mid-twentieth century was formulated by Heider in the 1940s [5, 10]. Each vertex of the triangle has a positive or negative relationship with the other two vertices. To judge the current status of the triangle, we first pick up the signs of the three edges (positive be “1”, negative be “-1”), then multiply the three signs. If the result is “1”, then the triangle is balanced. Otherwise, the triangle is unbalanced. As is shown in Fig. 44.1, triangles with three positive signs (*T3*) or two negative signs (*T1*) tend to be balance. On the contrary, triangles *T2* or *T0* tend to be unbalanced.

44.2.2 Symbols and Formulas

Social network is denoted by *S*. There are millions of nodes in the social network *S*; *N* is the set of the nodes.

Both *S1* and *S3* are subnets in which there are nodes satisfying the case *T1* or *T2* in Fig. 44.1. For example, taking two nodes *n*, *n'* from *S3*, we can find a third node *n''* in *S3* which satisfies that the signs of the edges between these nodes are “+”.

In Eq. (44.1), *E* is an adjacency matrix which is established according to the positive links in the data set. The elements in *E* represent the edges with a sign of “+”. We use *e_{xy}* to describe the elements of the Matrix *E*, the subscripts of *x* and *y* denote the row and the column of the elements. Such as, the element *e₁₅* in *E* indicates that the sign of the edge from node 1 to node 5 is “+” and node 1 trusts node 5.

$$\mathbf{E} = \begin{matrix} & \begin{matrix} n1 & n2 & n3 & n4 & n5 & n6 \end{matrix} \\ \begin{matrix} n1 \\ n2 \\ n3 \\ n4 \\ n5 \\ n6 \end{matrix} & \begin{pmatrix} 1 & & & & 1 & \dots \\ & 1 & 1 & & & \dots \\ 1 & 1 & & & 1 & \dots \\ & & & 1 & 1 & \dots \\ & & & & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \end{matrix} \tag{44.1}$$

In Eq. (44.2), E' is also established according to the negative links in the data set. The elements in E' represent the edges with a sign of “-”. We use e'_{xy} to describe the elements of the Matrix E' . For instance, the element e'_{21} in E' indicates that the sign of the edge from node 2 to node 1 is “-” and node 2 distrusts node1.

$$\mathbf{E}' = \begin{matrix} & n1 & n2 & n3 & n4 & n5 & n6 \\ \begin{matrix} n1 \\ n2 \\ n3 \\ n4 \\ n5 \\ n6 \end{matrix} & \begin{pmatrix} & & & & & \dots \\ -1 & & & & -1 & \dots \\ & & & & -1 & \dots \\ -1 & & & & & \dots \\ & & & & -1 & \dots \\ & & -1 & & & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} & \end{matrix} \quad (44.2)$$

44.3 Algorithm Analysis

The core of the algorithm is to extract the nodes which satisfy balance theory, and then we build a trust network according to the nodes and the relationships between them. Considering of the directions and the signs of the edges, we divided the analysis into two kinds of triples $T1$ and $T3$. We can see by the above description, we need to find out the triples satisfy the rules in Fig. 44.1 for $T1$ and $T3$.

44.3.1 Case of $T3$

In order to find the triples in the data set which satisfy the rules of $T3$, we divide the search into four cases according to both the sign and the direct of the edges in the triples. The four cases are as shown in Fig. 44.2.

The main aim here for picking out the triples of case $T3.a$ ($T3.d$) in Fig. 44.2 is to take two elements from elements whose values are “1” for each row (column) in E . Then we write down the coordinates of the elements, denoted as $(x1, y1)$, $(x2, y2)$. At last we determine whether the value of e_{y1y2} or e_{y2y1} is “1”. If the value is “1”, then we add this triple to the set $S3$. Circulate the steps above until all the probabilities have been tested. For case $T3.b$ and case $T3.c$ we process the diagonal elements of E . For each element (x, x) , we take two elements, one from the x th row whose value is “1”, the other from the x th column with the value of “1”. The coordinates of the two elements are $(x, y1)$, $(x1, x)$. We aim to check whether the value of element $(x1, y1)$ or $(y1, x1)$ is “1” in E . After all the possible elements are processed, we get the subnet $S3$.

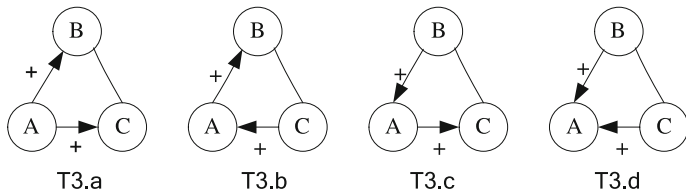


Fig. 44.2 Four cases of triples with directed edges for $T3$

44.3.2 Case of $T1$

We can see by the above description, taking into account of the sign and the direct of the edges, we also divide the search process into four cases for $T1$ as shown in Fig. 44.3.

As the number of edges with a sign of “-” is far less than the sign of “+”, so we do searching on E' . To extract the triples of case $T1.a$ ($T1.d$) in Fig. 44.3, first we choose two elements $(x, y1)$ and $(x, y2)$ with the value of “-1” from every row (column) of E' . Then determine whether the value of element $(y1, y2)$ or $(y2, y1)$ is “1”. If it satisfies, join the triple to $S1$. Similar with the case $T3.a$, circulate the steps. For case $T1.b$ and $T1.c$ we need to deal with each of the diagonal elements (x, x) in E' . We choose one element $(x, y1)$ whose value is “-1” in the x th row from E' and the other $(x1, x)$ from the x th column. If the value of the element $(x1, y1)$ is “1” in E , add the triple to $S1$. At last, we need to take two elements whose value is “-1” from every column of E' . The remaining steps are the same with case $T1.a$.

Note that the results of $S1$ and $S3$ contain redundant nodes and edges. Thus, we need to eliminate the redundancies when join the triples to $S1$ or $S3$. Then, we combine the set $S1$ with $S3$. Finally, we get the trust network.

44.4 Experiments

We make experiments on Epinions with 130,000 nodes and 840,000 edges. In the data set, there are 710,000 edges with a sign of “+”, and 130,000 edges with a sign of “-”. We extracted the edges from the triples which satisfy the rules of $T3$ or $T1$

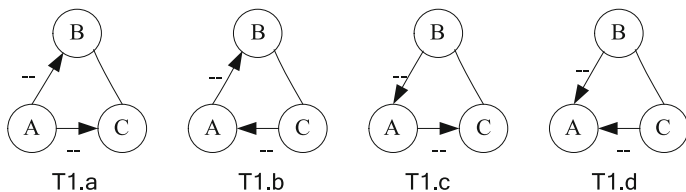


Fig. 44.3 Four cases of triples with directed edges for $T1$

Table 44.1 The comparisons between case *T3* and case *T1*

| Triple <i>Ti</i> | Number |
|------------------|-----------|
| <i>T3_a</i> | 3,206,203 |
| <i>T3_b</i> | 2,025,624 |
| <i>T3_c</i> | 3,207,569 |
| <i>T3_d</i> | 3,198,077 |
| <i>T1_a</i> | 296,080 |
| <i>T1_b</i> | 68,562 |
| <i>T1_c</i> | 271,534 |
| <i>T1_d</i> | 52,385 |

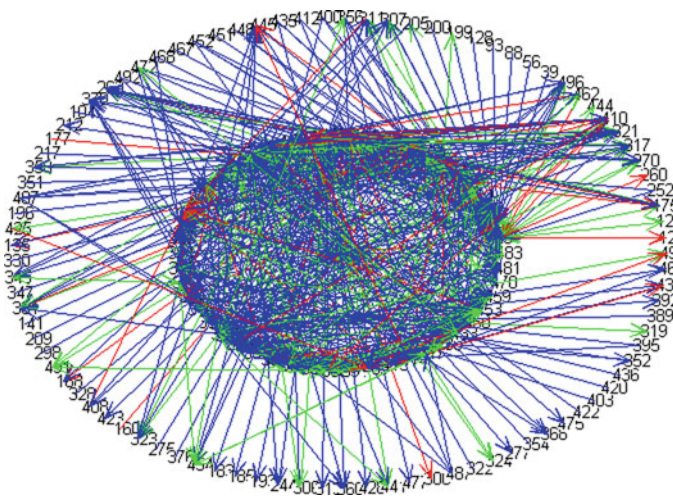


Fig. 44.4 Trust network graph which consists of more than 500 nodes

from the data set, and count up the number of different types of triples. The results are shown in Table 44.1.

In Table 44.1, there is the number of the *T3* and *T1* triples we found. Therefore, we can use the results to build a trust network and compute the trust values based on this trust network. In the trust network, the nodes link to others positively or negatively indicating that they trust them or not, then the networks can be viewed as a directed graph. In order to demonstrate our results more intuitive, we draw a directed graph according to the experimental data in Figs. 44.4 and 44.5.

As shown in Fig. 44.4, the graph is consisted of 500 nodes with directed edges. The nodes are extracted from the experiment results. In the graph, the blue lines represent the edges with the sign of “+” and the red lines represent the edges with the sign of “-”. If there are two nodes with different signs, we use green line to connect them. We also use more nodes to paint a huge trust network diagram as shown in Fig. 44.5. The diagram consists of more than 5,000 nodes and the edges

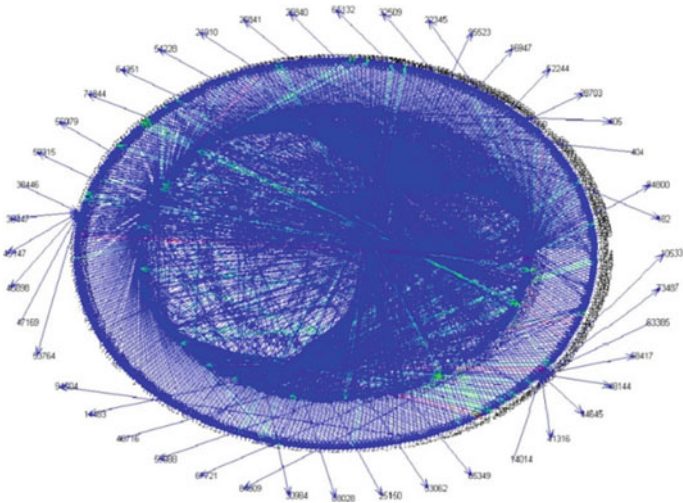


Fig. 44.5 Trust network graph which consists of more than 5,000 nodes

between them. This trust network diagram has six layers, we can see very few nodes in the layer 6 and edges there are also very sparse. This also confirms the six degrees of separation.

44.5 Conclusion

In this paper, we propose a new algorithm for building the trust network, which combines Heider’s balance theory. We make a series of experiments on Epinions data set. According to the experimental results, we analyze the ratio of different types of triples, and draw a trust network graph. In the next work, we will add some filter rules to the algorithm and do some trust inference based on the trust network.

Acknowledgments This work is supported by the Doctor Program Foundation of Education Ministry of China (No. 201110042120027), the Fundamental Research Funds for the Central Universities of China (No. N110417006, N110204003).

References

1. Wasserman S, Robins G, Steinley D (2008) Social network analysis: methods and applications, vol 7. Cambridge University, Cambridge, pp 90–98
2. Wu XW, Xu FY, Song W (2005) Some implications from competitive intelligence research based on social network. In: The China society for scientific and technical information 24:632–635

3. Tan ZH, Cheng W, Gao XX, Wang H, Chang GR (2008) A peer-to-peer overlay network routing protocol based on bidirectional circle topology. In: The 4th IEEE international conference on wireless communications, networking and mobile computing, vol 2, pp 1–4
4. Tan ZH, Cheng W, Ma Y, Chang GR (2009) An improved peer-to-peer routing algorithm K-CSSP based on communication history clustered by K-means, vol 2. In: The 9th international conference on hybrid intelligent systems, pp 381–385
5. Chen XF, Tan ZH, Yang GM (2011) A hybrid algorithm to solve traveling salesman problem. In: International conference on electronic engineering, communication and management. Lecture notes in electrical engineering, vol 139, pp 99–105
6. Kuter U, Golbeck J (2007) Sunny: a new algorithm for trust inference in social networks using probabilistic confidence models. In: National conference on artificial intelligence, vol 2, pp 1377–1382
7. Jøsang A, Hayward R, Pope S (2008) Optimal trust network analysis with subjective logic. In: 2008 second international conference on emerging security information, systems and technologies (Secureware), vol 6, pp 179–84
8. Jøsang A (1996) The right type of trust for distributed systems. In: Proceedings of the 1996 new security paradigms workshop, vol 9, pp 34–38
9. Ray I, Chakraborty S (2009) An interoperable context sensitive model of trust. *J Intell Inf Syst* 32:75–104
10. Heider F (1946) Attitudes and cognitive organization. *J Psychol* 21:107–112