Proceedings of the 2016 International Conference on Machine Learning and Cybernetics, Jeju, South Korea, 10-13 July, 2016

Link prediction based on dynamic weighted Social Attribute Network

RONG ZENG , YU-XIN DING, XIAO-LING XIA

Harbin Institute of Technology, Shenzhen, China
E-MAIL: yxding@hitsz.edu.cn

Abstract:

Social networks are highly dynamic objects; they grow and change quickly over time through the addition of new edges, signifying the appearance of new interactions in the underlying social structure. In present online social networks have become necessary communication tools in peoples' daily life. Links prediction of social network can not only recommend future friends to a user, but also can predict large scale social trends. In this paper, the link prediction problem is treated as a binary classification problem. We propose the method for constructing dynamic weighted social attribute network, and then extract different features from the weighted social attribute network, which are used to train a classifier for link prediction. Moreover, we discuss how to design the node weight and edge weight. The experiments show that when compared with the original social attribute network graph, the weighted social attribute network has better performance for link prediction, and the method for designing the weight of the node and edge weight of the social attribute network is feasible.

Keywords:

link prediction; social network; social-attribute network; weighted network; dynamic network

1. Introduction

Social networks are highly dynamic objects; they grow and change quickly over time through the addition of new edges, signifying the appearance of new interactions in the underlying social structure. In present online social networks have become necessary communication tools in peoples' daily life. So the researches of online social network have attracted much attentions in research communities. For example, a lot of works have been done on social network evolution [1] and network topology structure prediction, which are called Social Network Analysis [2]. Link prediction problem is also a fundamental problem in social network analysis that aims to infer which unobserved links will appear in the near future by a given snapshot of a network. In reality, link prediction has been widely used in many fields. In the field of e-commerce, link prediction technologies can be applied in the personalized product recommendation system and recommend users the products they are interested in [4]. In bioinformatics field, link prediction can be used to predict the interaction between proteins. Friends recommendation system in social networks is also a link prediction problem [5]. One special characteristic of social network is that it is an unbalanced network. The number of popular users on the social network is relatively small, but they always have tens of millions of fans. On the contrary, the number of common users is large and they only have a few friends. Thus friend recommendation service can provide much help for users to extend their social relations.

In this paper we propose the method for constructing dynamic weighted social attribute network, and then extract different features from the weighted social attribute network, which are used to train a classifier for link prediction. Moreover, we discuss how to design the node weight and edge weight. The experiments show that when compared with the original social attribute network graph, the weighted social attribute network has better performance for link prediction, and the method for designing the weight of the node and edge weight of the social attribute network is feasible.

2. Related works

Recently, a large number of approaches have been proposed to address the link prediction problem. The methods can be roughly classified into the following categories:

2.1. Methods based on the similarity score of the node pair

In these methods, a connection weight score(x, y) is assigned to pairs of nodes, and then the node pairs are ranked in decreasing order of score(x, y). Usually the higher the score value, the greater the possibility of the existence of links between nodes [3, 5]. A predictor can thus be seen as calculating a measure of similarity between

978-1-5090-0390-7/16/$31.00 ©2016 IEEE

nodes x and y, relative to the network topology. These predictors are adapted from techniques used in graph theory and social network analysis, and many must be adjusted from their original purposes to measure node-to-node similarity.

Usually two types of similarity measures can be used to compute the similarity of node pairs, local similarity and global similarity.

Local similarity: local similarity assigns scores to pairs of nodes according to node's neighbors, for example, we can use common neighbors, cosine similarity, Jaccard's coefficient, Adamic-Adar coefficients to calculate the similarity [3].

Global similarity: global similarity belongs to a kind of path-based method. It assigns scores to pairs of nodes according to the global network structure. Many algorithms based on random walk [6-8] have been proposed to compute global similarity.

2.2. Machine learning based methods

Machine learning based methods for link prediction methods can be classified into three categories: supervised methods, unsupervised methods and the semi-supervised methods.

The supervised method for link prediction predicts the unobserved links using a binary classifier. However, the supervised methods often suffer from the so-called class imbalance and feature selection problem [6]. Furthermore, most classifiers are based on the class distribution of the training data, thus they could have poor performance on the datasets that do not meet the prior assumptions. Instead, the unsupervised methods work in an agnostic way, thus they can naturally evade this problem. In addition, unsupervised methods do not need to decide which node features and edge features to use for link prediction, thus they also avoid the feature selection problem. For example, Lichtenwalter *et al* 10] proposed an unsupervised link prediction method, ProFlow. It is a constrained random walk method, which limit the number of steps in L steps, and thus the prediction results are not affected by the noise nodes far away from the origin. The method can be applied in the weighted or non-weighted, directed or undirected graph. In this paper, we focus on unsupervised methods for link prediction.

3. Method description

3.1. Mathematical formulas

Social Attribute Network [8] (SAN) is an augmented social graph with both attribute and structure information. There are two kinds of nodes in the Social Attribute Network, the user node and the attribute node. There are two kinds of links, the relationship between the user and the user, called social link; the relationship between the user and the attribute, called attribute link.

Construction of Social Attribute Network example:

Suppose there are three users, Ming, Lan and Hong. Their relationships and attributes are shown in Table 1. First we build a social network according to social relations among users, then we add the attributes of each used to each node. The Social Attribute Network is shown as Figure 1.

TABLE 1. Social Attribute Network information

User	Attribute	Follow
Ming	Man, C	Lan
Lan	Woman, Sing	Ming
Hong	Woman, Sing	Lan

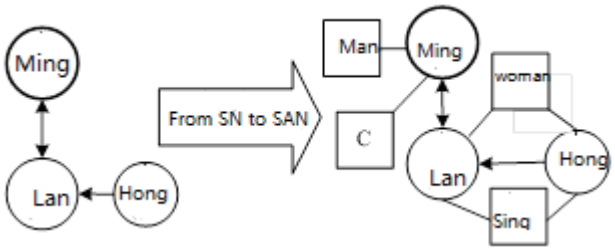

FIGURE 1. Social Attribute Network example

3.2. Weighted Social Attribute Network

The social attribute network is an unweighted graph. When we calculate the similarity of node pairs, all nodes and edges are equally considered. However, in fact different node and edges have different effect on the similarity of node pairs. To measure the effect of different nodes and edges on the similarity of node pairs, we assign weight values to each node and edge.

Assigning node weight. Node weight is to represent the contribution of different nodes for link prediction. We assign the node weight inversely proportional to degree centrality, shown as formula (1). The degree centrality of a node is equal to the number of the adjacent edges of the node.

$$w(v)=\begin{cases} e^{-\sqrt{d(v)-2}}, d(v)>1 \\ 1, otherwise. \end{cases} \quad (1)$$

Assigning edge weight. Edge weight represents the similarity of two entities [12]. In general, the edge weight between two users depends on the closeness of two user, the interaction frequency between them, the trust between each other and so on. We make the edge weight between two nodes proportional to the number of common neighbors of the two nodes, which is defined as formula (2).

$$w(x,y)=e^{-\sqrt{\max(CN)-CN(x,y)}} \quad (2)$$

In equation (2), CN(x,y) is the number of common neighbors between node x and y; max(CN) is the maximum number of common neighbors between all nodes.

3.3. Feature extraction in Dynamic Weighted Social Attribute Network

Next we extract features from undirected weighted social attribute network. Describing the social attribute network as am undirected graph means we only consider mutual friend relationships. The reasons are as follows: firstly, undirected graph is simple and two-way relations mean stronger and closer relations between them. Secondly, there are a lot of spammers who follow a large number of users, the use of two-way relations can also avoid the interference of spammers.

3.3.1. Features for local similarity

Common neighbors. The more common neighbor nodes two nodes have, the greater the similarity is. Formula (3) and (4) show the common neighbor feature defined on node weight and edge weight respectively. $F(u)$ represents the neighbors node u, and $w(t')$ represents the node weight of node t'. In equation (4), $w(u,t)$ represents the edge weight between node u and node t.

$$CN\_VW(u,t)=\sum_{t'\in F(u)\cap F(t)} w(t') \quad (3)$$

$$CN\_EW(u,t)=\sum_{t'\in F(u)\cap F(t)} (w(u,t')+w(t,t')) \quad (4)$$

Adamic-Adar coefficient. The Adamic-Adar coefficient [3] find out the common neighbors of node pairs, then add the importance degree of each common neighbors. The importance of a neighbor is inversely logarithmic proportional to its degree. Formula (5) and (6) show the Adamic-Adar coefficient feature defined according to node weight and edge weight respectively.

$$AA\_VW(u,t)=\sum_{t'\in F(u)\cap F(t)} \frac{w(t')}{\log(1+\sum_{z\in F(t')} w(z))} \quad (5)$$

$$AA\_EW(u,t)=\sum_{t'\in F(u)\cap F(t)} \frac{w(u,t')+w(t,t')}{\log(1+\sum_{z\in F(t')} w(z,t'))} \quad (6)$$

Cosine similarity. The cosine similarity [5] is the proportion of common neighbors in all neighbors of node u and node t. Formula (7) and (8) show the cosine similarity feature defined according to node weight and edge weight respectively, where d_u represents the degree of node u.

$$COS\_VW(u,t)=\sum_{t'\in F(u)\cap F(t)} \frac{w(t')}{\sqrt{d_u d_t}} \quad (7)$$

$$COS\_EW(u,t)=\sum_{t'\in F(u)\cap F(t)} \frac{w(u,t')+w(t,t')}{\sqrt{d_u d_t}} \quad (8)$$

Jaccard coefficient. Jaccard Coefficient [3] is equivalent to the normalized common neighbor. Formula (9) and (10) show the cosine similarity feature defined according to node weight and edge weight respectively.

$$JC\_VW(u,t)=\frac{\sum_{t'\in F(u)\cap F(t)} w(t')}{\sum_{t'\in F(u)\cup F(t)} w(t')} \quad (9)$$

$$JC\_EW(u,t)=\frac{\sum_{t'\in F(u)\cap F(t)} (w(u,t')+w(t,t'))}{\sum_{t'\in F(u)\cup F(t)} (w(u,t')+w(t,t'))} \quad (10)$$

3.3.2. Features for global similarity

Random walk with restart(RWR). Random walk with restart on weighted social attribute network (RWR_WSAN) can alleviate the sparsity problem. The walker randomly walks along the edge from a node to its adjacent node. When the walker goes to a node, the next step continue to walk at the probability of a, or the walker back to the start node at the probability of (1-a). The greater the probability a, the RWR feature gets more global information; otherwise, the smaller the probability a, the RWR feature gets more local information. The visiting probability $P(v_u,v)$ from the start node v_u to node v is calculated by formula (11). The

RWR feature is calculated by formula (12):

$$P(v_u,v)=\begin{cases}(1-a)+a\sum\limits_{v'\in F(v)}\dfrac{P(v_u,v')w(v',v)}{\sum\limits_{t'\in F(v')}w(v',t')},(v=v_u)\\ a\sum\limits_{v'\in F(v)}\dfrac{P(v_u,v')w(v',v)}{\sum\limits_{t'\in F(v')}w(v',t')},(v\neq v_u)\end{cases} \tag{11}$$

In formula (11), v_u is the start node of walker; a is the possibility of continue walking from a node; $P(v_u,v)$ is the possibility of visiting node v from start node v_u; $w(v,t)$ is the edge weight of edge (v,t).

$$RWR\_WSAN(v_u,v)=(P(v_u,v)+P(v,v_u))/2 \tag{12}$$

3.3.3. Temporal feature

The normal temporal feature is defined as the node degree changing with time [13]. If two nodes are more active in recent, then the probability that the two nodes will generate the link is greater; otherwise, the probability is smaller. In this way, we design the TIME feature as the sum of the number of new links between two nodes in a time interval, as the formula (13):

$$TIME(u,v)=d_t(u)-d_{t'}(u)+d_t(v)-d_{t'}(v) \tag{13}$$

In formula (13), $d_t(u)$ is the degree of node u at time t (t>t').

4. Experiments

4.1. Dataset description and performance evaluation

Dataset description. We do experiments on the Google+ dataset [11]. Google+ dataset collected the data of social network users from July, 2011 to September 2011. The description of the dataset is shown in Table 2.

TABLE 2. Google+ dataset information

	Social links	All social links	User node	Attribute links	Attribute node
JUL4	7062	7062			
AUG4	7430	7831	5200	24690	9539
SEP4	7422	8100			

The number 4 in July (JUL4), August (AUG4), September (SEP4) means that when an attribute node has at least 4 neighbors, it is considered to be an effective attribute node. The second column is the number of social links, the links between users. The third column is the sum of the second column and the number of hidden social links (if a social link exist in July, but lost in August, we call such link as a hidden link. We assume the link is set invisible to the public. in fact, hidden social links still exist).

Performance evaluation. There are four kinds of prediction results on the test dataset:

TP - number of node pair that will build a new link and the node pair is predicted to build a new link;

FN - number of node pair that will build a new link and the node pair isn't predicted to build a new link;

FP - number of node pair that will not build a new link and the node pair is predicted to build a new link;

TN - number of node pair that will not build a new link and the node pair isn't predicted to build a new link.

The dataset for social network link prediction is an imbalanced dataset. In general we use the Area under the Receiver Operating Characteristic Curve (AUC) to evaluate the performance of a method tested on an imbalanced dataset.

In the Receiver Operating Characteristic Curve (ROC), the longitudinal axis is true positive rate (TPR). TPR means the proportion of the positive samples that are correctly predicted as the positive samples by the classifier. TPR is defined as formula (14).

$$TPR=TP/(TP+FN) \tag{14}$$

In the ROC curve, the horizontal coordinates is the false positive rate (FPR). FPR means the proportion of the negative samples that are wrongly predicted of the whole negative samples by the classifier. TPR calculation formula is (15):

$$FPR=FP/(FP+TN) \tag{15}$$

In this paper, we use the AUC to evaluation the performance of the prediction model. When the AUC value is greater than 0.5, indicating that the classifier is effective; when the AUC value is 1, indicating that the performance of the classifier is best and is a perfect classifier.

4.2. Unsupervised experiments

4.2.1. Analysis of features for local similarity

In our experiments we use the unsupervised linear threshold classifier to predict links between users. We obtain the complete graph of social network, and the

experimental samples are the node pairs of the edges of the complete graph. If the node pair is an edge of social network, the node pair is a positive samples, otherwise, the node pair is a negative sample. In the experiments we only use local similarity features to predict links. If the feature value of a node pair is bigger than the threshold, the classifier predicts a link existed between the node pair. Otherwise, no links existed. The experimental results on the complete graph of social network are shown in Table 3. From Table 3, we can see the overall performance of features with edge is better than that of features with node weight, and the overall performance of features with node weight is better than that of features with no weight. Compare with other features, we can see that the local similarity feature, Adamic-Adar coefficient, perform the best.

TABLE 3. The AUC of local similarity feature of social network in Sep.

Feature	No weight [11]	Node weight	Edge weight
Common neighbor	0.72630	0.72640	0.72660
Adamic-Adar	0.72641	0.72639	0.72669
Cosine	0.72616	0.72626	0.72662
Jaccard	0.72620	0.72627	0.72662

Table 4 shows the results of the local similarity features on the social attribute network. From Table 4, we can get the similar result as that from Table 3.

TABLE 4. The AUC of local similarity feature of social attribute network in Sep.

Feature	No weight [11]	Node weight	Edge weight
Common neighbor	0. 82978	0. 83273	0. 83212
Adamic-Adar	0. 83281	0. 83266	0. 83434
Cosine	0. 82845	0. 83228	0. 83202
Jaccard	0. 82835	0. 83238	0. 83195

From the experimental results, it shows the performance of features with node weight is better than that of features with no weight in general, and the performance of features with edge weight has the best performance. So feature extraction method discussed in section 3.2 can improve the accuracy of link prediction.

4.2.2. Analysis of features for global similarity

The step of walker is taken as 20. The social network dataset of July is denoted as SN. We construct the fully connected graph of SN, and the node pairs of edges of fully connected graph is represented as the set A. We represent the node pairs of edges of SN as set P. The set U stores node pairs of edges (A-P). The set U is used as the test dataset. If a node pair in U creates a new link in the dataset of August, the node pair is a positive sample, otherwise the node pair is a negative sample. We extract the feature, random walk with restart with edge weight and without edge weight, from the social attribute network, respectively. Then we make link prediction using unsupervised linear threshold classifier. The experimental results are shown in Figure 2 and Figure 3. Figure 2 is the result of the random walk feature extracted from the social network. Figure 3 is the result of the random walk feature extracted from the social attribute network. We can see that the performance of the feature with edge weight is better than that of the feature without edge weight. The horizontal axis represents the value of the start possibility *a*. We can see when the start possibility equals to 0.93, we get the best AUC 0.91957.

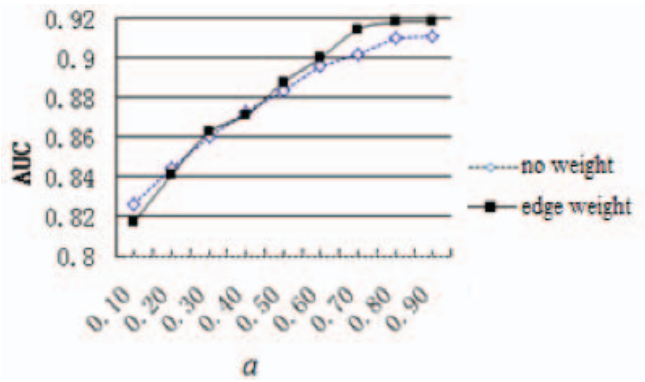

FIGURE.2. The influence of *a* on the result of random walk with restart of the social network

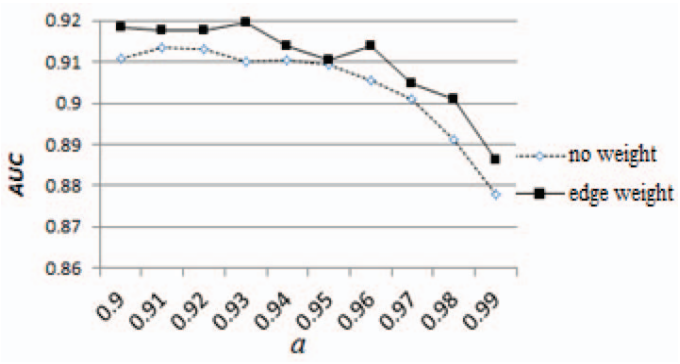

FIGURE 3. The influence of *a* on the result of random walk with restart of the social attribute network

4.2.3. Analysis of temporal feature

We use the social network snapshots in August and September as the testing dataset. We also use the method in section 4.2.2 to define the positive and negative samples. We extract temporal features from the social network and social attribute network, respectively. We make link prediction using unsupervised linear threshold classifier, and the AUC results using the two types of temporal features are both 0.7259, which means that the temporal feature is effective for link prediction.

5. Conclusion

In this paper, the link prediction problem is treated as a binary classification problem. We propose the method for constructing dynamic weighted social attribute network, and then extract different features from the weighted social attribute network, which are used to train a classifier for link prediction. We discuss how to design the node weight and edge weight. The experiments show that when compared with the original social attribute network graph, the weighted social attribute network has better performance for link prediction, and the method for designing the weight of the node and edge weight of the social attribute network is feasible.

Acknowledgements

This work was partially supported by Scientific Research Foundation in Shenzhen (Grant No. JCYJ20140627163809422), Guangdong Natural Science Foundation (Grant No. 2016A030313664) and Key Laboratory of Network Oriented Intelligent Computation (Shenzhen).

References

[1] Yan F, Zhang M,. Evolution of Social Networks: New Patterns and a New Generator, Proceedings of the Creating, Connecting and Collaborating through Computing (C5), 2011 Ninth Conference on: IEEE, 2011: 3-10.

[2] Jamali M, Abolhassani H. Different Aspects of Social Network Analysis,Proceedings of the Web Intelligence 2006 International Conference on : IEEE, 2006: 66-72.

[3] Liben-Nowell D, Kleinberg J. The Link-prediction Problem for Social Networks[J]. Journal of the American Society for Information Science and Technology, 2007, 58(7): 1019-1031.

[4] Corlette D. Link Prediction Applied to an Open Large-Scale Online Social Network, Proceedings of the Proceedings of the 21st ACM Conference on Hypertext and Hypermedia: ACM, 2010: 135-140.

[5] Leicht E. Vertex Similarity in Networks[J]. Physical Review E, 2006, 73(2): 1-10.

[6] Backstrom L. Supervised Random Walks: Predicting and Recommending Links in Social Networks, Proceedings of the Proceedings of the Fourth ACM International Conference on Web Search and Data Mining: ACM, 2011: 635-644.

[7] Li R-H. Link Prediction: the Power of Maximal Entropy Random Walk, Proceedings of the Proceedings of the 20th ACM International Conference on Information and Knowledge Management: ACM, 2011: 1147-1156.

[8] Yin Z, Gupta M, et al. A Unified Framework for Link Recommendation Using Random Walks, Proceedings of the Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on : IEEE, 2010: 152-159.

[9] Newman M E. Clustering and Preferential Attachment in Growing Networks[J]. Physical Review E, 2001, 64(2): 1-13.

[10] Lichtenwalter R N. New Perspectives and Methods in Link Prediction, Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: ACM, 2010: 243-252.

[11] Gong N Z. Joint Link Prediction and Attribute Inference Using a Social-Attribute Network[J]. ACM Transactions on Intelligent Systems and Technology, 2014, 5(2): 1-20.

[12] Newman M E. Analysis of Weighted Networks[J]. Physical Review E, 2004, 70(5): 056131.

[13] Potgieter A. Temporality in Link Prediction: Understanding Social Complexity[J]. Emergence: Complexity & Organization (E: CO), 2009, 11(1): 69-83.