Knowledge-Based Systems 98 (2016) 200–215

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Prediction of missing links based on community relevance and ruler inference

Jingyi Ding*, Licheng Jiao, Jianshe Wu, Fang Liu

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an, Shaanxi Province 710071, China

ARTICLE INFO

Article history:
Received 31 July 2015
Revised 22 January 2016
Accepted 23 January 2016
Available online 2 February 2016

Keywords:
Link prediction
Community detection
Community relevance
Ruler inference
Complex networks

ABSTRACT

The link prediction algorithm which based on node similarity is the research hotspot in recent years. In addition, there are some methods which based on the network community structure information to predict the missing links, however, these studies only concerned about the obvious information between different communities such as direct links. We found that it is hard to predict the missing links if the two communities have little direct connections. In fact, there is similarity between communities such as the similarity between nodes and this similarity is significant for prediction. So, we define a community similarity feature which named community relevance by using not only the obvious information but also the latent information between different communities in this paper. Then a novel algorithm which based on the community relevance and ruler inference is proposed to predict missing links. In this method, we extract the community structure by using the local information of the network first. Next, calculate the relevance of each pair of communities by using the new community relevance indices. Finally, a simple prediction model which based on ruler inference is applied to estimate the probability of the missing links. It is shown that the proposed method has more effective prediction accuracy and the community relevance features improve the predictor with low time complexity, with experiments on benchmark networks and real-world networks in different scales, and compared with other ten sate of the art approaches.

1. Introduction

A network is composed of different individuals and their static or dynamic relations. These relations can be friendship among people or physical interaction among genes. Link prediction, predicting missing links or future structure of the network from the observed structure, is an important task in link mining [1,2], it is also an effective method and means for data mining. Link prediction helps us not only to understand the evolution mechanism of the complex networks theoretically [3,4,5], but also to solve very important issues in applications. For example, it can be used in the field of information integration, social network analysis, recommender systems [6] and bioinformatics.

We can divide the existing methods for link prediction into three categories. The first kind of methods define a measure of similarity between two nodes in the network, considering that links between two nodes with more similar are of higher existing probability [7,8]. The second kind of approaches based on the maximum likelihood estimation [9]. The third kind of algorithms based on machine learning techniques [10,11]. There are also many studies related to link prediction focusing on more complicated networks, like directed and weighted networks [12,13].

Link prediction in networks is one of the most important issues in complex networks. Previous proposed algorithms are mainly based on Markovian chains and machine learning. Then, a prediction method based on node similarity was proposed in 1998 [7]. Owing to the low complexity and high prediction accuracy, the link prediction algorithm which based on node similarity is the research hotspot in recent years. After that Nowell and Kleinberg summarize many similarity measures in ref. [8]. One kind of measures based on node neighborhoods, which concern the local structure of the networks. Another based on all paths of a network, which consider the global structure of the networks. However, the first kind of measures may not be enough for edge prediction, and the second measures have a high computational complexity. Afterwards, it's found that the hierarchical organization information [14,15] may indeed provide insights for link prediction. So, a maximum likelihood method which considered the hierarchical organization information of the network was proposed by Clauset,

* Corresponding author. Tel.: +86 15929737040.
E-mail address: jyding87@163.com, dingfeimn@163.com (J. Ding).

http://dx.doi.org/10.1016/j.knosys.2016.01.034
0950-7051/

Table 1
The advantages, the disadvantages and the time complexity of the algorithms.

Algorithm	Advantages	Disadvantages	Time complexity
CN/JC	(1) Low complexity	(1) Treat each common neighbor equally (2) General accuracy	$O(N^2)$
AA/RA	(1) Differentiate the contributions of different neighbors (2) Low complexity	(1) General accuracy	$O(N^2)$
CAR_CN/JC/AA/RA	(1) Consistent robustness (2) Proposed a local-community paradigm	(1) Network structure constraint	$O(N^2) \sim O(N^4)$
Yan's algorithm	(1) Using the community structure information	(1) Network structure constraint (2) The complexity of community division method has great influence on the whole algorithm	It takes $O(N^2)$ to calculate the probability of missing links
HSM	(1) Uncovers the hidden hierarchical organization	(1) Network structure constraint (2) High time complexity	In the worst case, it takes exponential time to sample dendrograms
SBM	(1) Identify possible spurious links, (2) Network reconstruction; (3) Robust (4) Flexible	(1) High time complexity	In the worst case, it takes exponential time to sample different partitions[a*]
SPM	(1) Robust (2) Consistent (3) Proposed a universal structural consistency index	(1) High time complexity	$O(N^3)$[b*]
CP	(1) Using the multi- resolution information of the network (2) Uncovers the hidden community structure	(1) Network structure constraint (2) The hub nodes and the clustering coefficient constraints	$O(kN^2)$
CRCN/CRJC/CRAA/CRRA	(1) Define the community relevance feature (2) Uncovers the hidden community structure (3) Effective (4) Efficient	(1) Network structure constraint (2) Degree distributions constraint	$O(N^2)$

[a*] The number of distinct partitions of N elements into groups is $\sum_{k=1}^{N} \frac{1}{k!} \sum_{l=1}^{k} \binom{k}{l} (-1)^{k-l} l^N$, which grows faster than any finite power of N [16].
[b*] The time complexity of computing the eigenvalues of the matrix is $O(N^3)$.

Moore and Newman in 2008 [9]. This algorithm combined with Monte-Carlo algorithm and fit a hierarchical model on a network graph to make link predictions. The results showed that the performance is well if the network has obvious hierarchical organization. However, the disadvantage of the algorithm is the runtime required increase exponentially as the number of vertices increases in the worst case. Next, another representative method which named stochastic block model was proposed by Guimerà in 2009 [16]. In the stochastic block model, nodes are clustered into groups and the probability that two nodes are connected depends only on the groups to which they belong. The advantages of this method are that we can identify not only missing interactions but also spurious interactions. Besides, we can also generate a reconstructed network from a single observed network by using this model. Unfortunately, the algorithm is very time consuming because the number of different partitions of N elements grows faster than any finite power of N. The time complexity of the method increase exponentially as the number of vertices increases in the worst case. Later, a method based on the community structure was proposed by Yan in 2012 [17]. It's proved that the community structure information is of great significance for link prediction. In the paper, the missing links are divided into two different parts, links in the same community nodes and links between different communities nodes, they tried to rank them by using the same node similarity indices, considering that links in the same community nodes are more similar and have higher existing probability. From this method, we can find that the existing probability of the missing links is zero when the similarity of two terminal nodes which in different communities is zero. This is different from the real phenomenon. Afterwards, Cannistraci et al. proposed a new method which based on the strategy of link-community in 2013 [4]. His algorithm introduces a new philosophy in the formulation of neighborhood-based indices which is named CAR-based variant. CAR suggests that two nodes are more likely to link together if their common-first-neighbors are members of a strongly inner-linked cohort. The time complexity of the method is $O(N^2) \sim O(N^4)$. Next, an algorithm based on the multi-resolution community structure information of the network was proposed by Ding et. al. in 2014 [18]. Instead of the accurate classification result obtained by a general community detection algorithm, the proposed method just needs the results obtained under different resolutions. The time complexity of the method is $O(kN^2)$. However, the performance of the algorithm is not good enough when the networks have too many hub nodes or the clustering coefficient is too large. In 2015, Lü et. al. proposed a structural perturbation method for link prediction that is more accurate and robust [19]. Their fundamental assumption is that missing links are difficult to predict if their addition causes huge structural changes, and thus network is highly predictable if the removal or addition of a set of randomly selected links does not significantly change the network's structural features. The advantage of this method is that its predictions are consistently more robust. The disadvantage is high time complexity. The time complexity is $O(N^3)$.

Then, we summarize the advantages, the disadvantages and the time complexity of the algorithms and show them in Table 1.