A Bayesian Method for Link Prediction with Considering Path Information

Suyuan Zhang[1], Lunbo Li[1(✉)], Cunlai Pu[1], and Siyuan Zhou[2]

[1] School of Computer Science and Engineering,
Nanjing University of Science and Technology, Nanjing 210096, China
lunboli@163.com, pucunlai@njust.edu.cn
[2] College of Computer and Information, Hohai University, Nanjing 210098, China

Abstract. Predicting links among nodes in the network is an interesting and practical problem. Many link prediction methods based on local or global topology alone have been proposed. There is a need to combine these two types of methods to further improve the prediction performance. In line with this direction, we study the link prediction problem based on the Bayesian method and propose a new link prediction method, i.e., path-based Bayesian (PB) method. In this prediction method, we give the definition of clustering coefficients of paths and use it to quantify the contribution of paths to link generation. Then, we propose a new link prediction method by combining the clustering coefficient of paths and Bayesian theory. Simulation results on real-world networks show that our prediction method has higher prediction accuracy than the mainstream methods.

Keywords: Link prediction · Path information · Bayesian method · Structural similarity

1 Introduction

There are countless networks in real world, such as social networks, communication networks, and transportation networks [1]. However, the part of these networks that we can directly observe is usually incomplete. If we use scientific methods to detect the complete network structure, it will consume a lot of resources. Under this circumstance, link prediction is getting more and more attention, since its aim is to predict missing links based on incomplete information with low cost [2]. For example, in protein networks, link prediction method proposed in Ref. [3] is able to suppress the indirect interactions in proteins and further reveal unknown direct relations in the network. In social networks, link prediction methods help to explore the relations among individuals. Friend recommendation system is built on this basis. In recent year, a lot of link prediction methods have been proposed [4], but the prediction performance of these methods still have a large room to improve. So we did some research in this paper to get higher prediction accuracy.

X. B. Zhai et al. (Eds.): MLICOM 2019, LNICST 294, pp. 361–374, 2019.
https://doi.org/10.1007/978-3-030-32388-2_31

The common assumption of link prediction is that if two nodes are similar, they have some tendency to be connected. Existing link prediction studies fall into two categories. One of them considers structural similarity and further design prediction methods. Structural similarity is determined by the network topology. The other applies machine learning methods to link prediction problems. Machine learning methods quantify the similarity of nodes based on their attributes, such as job, gender, hobby, etc.

The structural similarity based methods use information of local, global, or semi-local topology to make predictions. Link prediction methods based on common neighbors, such as common neighbor (CN) index [5], Adamin-Adar (AA) index [6], and Resource Allocation (RA) index [7], rate node pairs with common neighbors shared by them. It is consistent with the fact that the more common friends people have, the more likely they are to be introduced as friends. Different similarity standards describe different interaction modes of nodes in the network. Link prediction methods based on paths, e.g., the Katz index [8], are convinced that paths promote link generation in networks and weights of paths of different lengths are different. Link prediction method based on random walk can help us analyze the flow of information when the network structure is too complicated. In order to predict the evolutionary trend of the overall structure of the network, a structural perturbation method [9] is proposed for link prediction. This method applies the perturbation method which is used to determine the structural consistency to predict missing links.

Machine learning based link prediction methods have wide coverage, including Bayesian method [10], Markov chain [11], deep learning [12], etc. These methods mainly determine whether links are going to be generated or not based on node attribute information. For the first time, Liu et al. [13] applied a naive Bayesian model to link prediction problem. This algorithm takes the posterior probability of the link generation as the score of the link. If more network information is added, such as the degree of common neighbors, the accuracy of the algorithm can be further improved. Deep learning method is capable of understanding complicated networks. Wang et al. [14] proposed a hierarchical Bayesian deep learning method, which comprehensively considers high-dimensional attribute information and link structure with hidden variables. This algorithm makes full use of information and improves prediction accuracy. Although machine learning based link prediction method obtains good prediction results and is able to handle the cold start problem, its learning process is sometimes too complicated to complete in a certain period of time. Besides, some real information is difficult to obtain in practice.

As mentioned before, one of the mainstream research directions for link prediction is to make full use of network topological information. However, existing topology-based link prediction algorithms still have room for further research. In the existing prediction methods, paths of the same length are generally considered to be equivalent, but even paths of the same length have different structure. So their contribution to the formation of links should be different. On the other hand, the relative importance of paths of different lengths should be determined by the actual network topology, but there is currently no acknowledged standard to determine the relative importance of paths of different lengths in link prediction. Existing methods use only local topology information or global topology information for link prediction. Few methods consider multiple types of information at the same time. The actual score of a node pair is not determined by a certain factor alone. It is affected by the local topology, the global topology, and the evolution trend of the network at the same time.

In order to solve these problems and obtain more accurate prediction results, this paper proposes a path-based Bayesian method for link prediction based on network topology. Specifically, in order to determine the contribution of a single path, a statistical method is used to generalize the clustering coefficient of nodes to the clustering coefficient of paths, and then use it to quantify the contribution of paths to link generation. On this basis, we use Bayesian theory to further revise the contribution of paths which makes the score of paths more reasonable. The sum of paths scores between nodes is proportional to the possibility of link generation. Experiments are carried out in the real network to test the performance of the algorithm. The main contributions of this paper are as follows:

- First, inspired by the clustering coefficient of nodes, this paper proposes the clustering coefficient of paths to quantify the ability of paths to facilitate links in networks. Through the statistics of local topological information around a path, we can get the clustering coefficient of paths. Only after that can we calculate the prior contribution of paths.
- Second, Bayesian method converts prior contribution into posteriori contribution. In the process of transformation, global topological information is introduced which makes the posteriori contribution more reasonable. By using Bayesian method in different dimensions, we can get the relative weights of different length paths in the network. Based on this idea, this paper proposes a path-based Bayesian method for link prediction.
- Third, we tested the proposed algorithm in several real networks with AUC and Precision indexes. Results of the experiments show that the proposed method performs better than traditional algorithms. Afterwards, the results of experiments and the causes are discussed in detail. The results of the experiments prove the availability of our algorithm.

The rest of this paper is organized as follows. In Sect. 2, we define the clustering coefficient of paths according to the actual topological structure of networks, so that it can be used to define the prior contribution of paths. In Sect. 3, we describe Bayesian theory and discuss its applicability in link prediction problems. In particular, we analyze why Bayesian theory can use global topological information to estimate the weights of paths of different lengths. Combining known information of current networks, we propose a path-based Bayesian method for link prediction and show the process of realizing this method in Sect. 4. Section 5 introduces the indexes which are going to be tested in the following experiments. Then, we show the results of link prediction methods in several real world networks. These results and their causes are discussed in Sect. 6. In the end, conclusion is made in Sect. 7.

2 Clustering Coefficient of Paths

Among the existing link prediction research, methods based on local information use little information and have limited prediction accuracy. On the other hand, methods based on global information are computationally complex. Methods based on paths guarantee prediction accuracy to a certain extent and are easy to calculate. Therefore, in a homogeneous network, it is reasonable to use a link prediction method based on paths.

In the design of link prediction algorithms, it is necessary to quantify the contribution of the paths to link generation. We use this contribution as a score for node pairs to predict links in the network. In order to get higher prediction accuracy, the score of each path needs to reflect its ability to facilitate links in the network. Inspired by the definition of the clustering coefficient of nodes [15], we define the clustering coefficient of paths (CCP) and use it to quantify the contribution of a single path. The clustering coefficient of the path is defined as the probability that the core of paths contributes to the link. As shown in the Fig. 1, in order to calculate the clustering coefficient of the path, we define the embedded path ω_{pq} of the path ω_{xy} as the core of the path ω_{xy}. In the case where the path length is 2, the node p coincides with the node q.

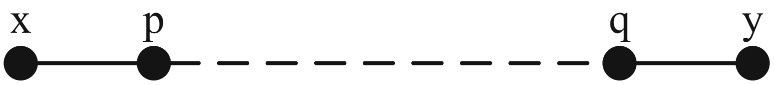

Fig. 1. Embedded path ω_{pq} is the core of path ω_{xy}.

In the case where a path of length k is known to exist, statistical methods are applied to calculate its contribution to facilitate a link between its source node and its destination node. At first, we assume that the number of links that were successfully contributed by the path, which is represented as N_L, and the number of links that were not successfully contributed, which is represented as N_U, are all 1. As shown in the Fig. 2, starting from the core of paths, the neighbor combinations of the core (x_i, y_j) is going to be detected. For each combination, if it is connected, the number of links which are successfully contributed by this path is increased by 1 and vice versa. The clustering coefficient of paths is the probability that the core of paths contributes to the link, which can be represented as

$$CCP = \frac{N_L}{N_L + N_U}. \tag{1}$$

Since one core serves multiple paths at the same time, the clustering coefficient of the path need to be further modified based on the path structure to quantify the contribution of a single path to the link generation. Inspired by the RA index, we use the degree of the nodes at both ends of the core to modify the clustering coefficient of the path. The greater the degree the nodes at both ends of the core have, the smaller contribution the path own. After modification, the prior contribution of a single path $P(A_1|\omega_k)$ is

$$P(A_1|\omega_k) = \frac{2}{k(p) + k(q)} * CCP, \tag{2}$$

where A_1 indicates that the link exists. ω_k represents a path of length k. $P(A_1|\omega_k)$ describes the possibility that a path of length k facilitate a link. $k(p)$ and $k(q)$ are degrees of node p and node q.

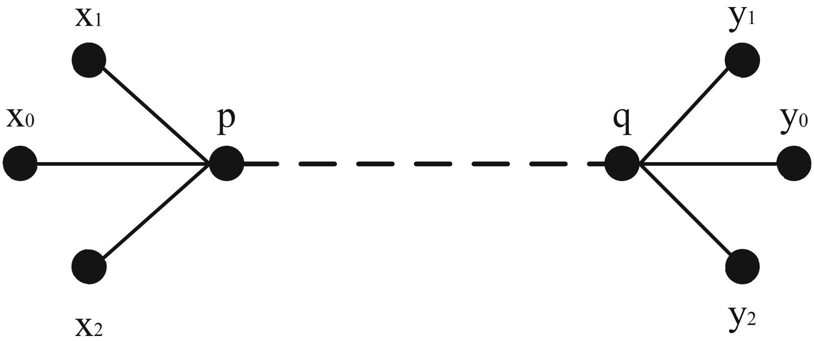

Fig. 2. (x_i, y_j) represents the neighbor combinations of the core of path.

3 Bayesian Method

After calculating the contribution of a single path, it is also necessary to use the network information to determine the relative weights of the paths of different lengths. The traditional link prediction method considers that the longer the path, the smaller the contribution it has. However, length of the path is not the only determinant. If path of a certain length appears more in the network, it means this kind of path plays important role in the network communication. As a result, its weight should be greater. To distinguish the importance of paths of different lengths, we use the Bayesian method. This method modifies the prior contribution $P(A_1|\omega_k)$ of paths with global topology information to obtain conditional probability $P(\omega_k|A_1)$, which is

$$P(\omega_k|A_1) = \frac{P(\omega_k)}{P(A_1)} * P(A_1|\omega_k), \tag{3}$$

where $P(A_1)$ indicates the existing probability of the link between target node pair (a,b). Since the node pairs which we want to predict are not connected, it is impossible to calculate the existing probability of link directly. We ignore the differences between links and take the probability of links appearing in the network as $P(A_1)$, i.e., $P(A_1) = N(A_1)/N_{max}(A_1)$. $N(A_1)$ represents the number of links which exist in the network. $N_{max}(A_1)$ represents the maximum number of links that may appear in the network. $P(\omega_k)$ represents the probability of a path of length k existing between two nodes. Similarly, it is processed in the same way as $P(A_1)$, i.e., $P(\omega_k) = N(\omega_k)/N_{max}(\omega_k)$. $N(\omega_k)$ represents the number of paths of length k which exist in the network. $N_{max}(\omega_k)$ represents the maximum number of paths of length k that may appear in the network. This method of processing data can effectively reduce the computational complexity.

At the same time, since the network rarely mutates, the existing probability of the path of length k in the network will fluctuate around its expected value $E(P(\omega_k))$ for a period of time. The more active the path of length k, the higher the expected value. The expected value reflects the evolution trend of the current network, so we need to use it to make another correction to the contribution of the path. The true contribution of a single path will be corrected as

$$P'(A_1|\omega_k) = \frac{E(P(A_1))}{E(P(\omega_k))} * P(\omega_k|A_1), \tag{4}$$

where $P'(A_1|\omega_k)$ indicates the posterior contribution of a known path of length k to link generation between a node pair. $E(P(A_1))$ represents the expectation of $P(A_1)$. Similarly, $E(P(\omega_k))$ represents the expectation of $P(\omega_k)$. In the absence of additional information, we believe that the expectation of $P(\omega_k)$ is a ratio of the maximum possibility. At the same time, the expectation of paths of different lengths are assumed to be the same, i.e., $E(P(\omega_k)) = \lambda$. This parameter can be further calculated when more information is available for reference.

4 Our Method

In a network $G(V, E)$, V represents nodes in the network, and E represents relationships between nodes. The number of nodes in the network is represented by $|V| = N$, and the number of edges is $|E| = M$. If we only have a single-layer network, the expected number of paths of length 1 is the same as the number of links which actually exist, that is $E(P(A_1)) = P(A_1)$. In a single-layer network, we suppose that $E(P(\omega_k))$ is a fixed ratio of the maximum value of $P(\omega_k)$ at any k value. In this case, the posterior contribution of the path can be simplified as

$$P'(A_1|\omega_k) = \frac{P(\omega_k)}{*} P(A_1|\omega_k), \tag{5}$$

where $P(\omega_k)$ indicates the existing possibility of paths of length k. The more active the path of this length is, the more important it is in the network. As a result, its weight should be greater. Paths of lengths 2 and 3 in the network are highly efficient in transmitting information due to their short length. These paths dominate the path algorithm, so we only need to calculate $P(\omega_2)$, and $P(\omega_3)$. We can know from previous discussion: $P(\omega_2) = N(\omega_2)/N_{max}(\omega_2)$, and $P(\omega_3) = N(\omega_3)/N_{max}(\omega_3)$. $N_{max}(\omega_2)$ represents the maximum number of paths of length 2 that may appear in the network, which is

$$N_{max}(\omega_2) = \begin{cases} M(M-1) & M \leq N \\ n_2 L_{2max} + 2m_2 n_2 + \dfrac{m_2(m_2-1)}{2} & M > N, \end{cases} \tag{6}$$

where L_{2max} means the max number of paths of length 2 that can be created by one node, so $L_{2max} = (N-1)(N-2)$. n_2 is the maximum number of the nodes in the network that connect to all other nodes. m_2 represents the number of edges remaining after n_2 nodes connect to all other nodes, $m_2 = M - n_2(N-1) + n_2(n_2-1)/2$. $N_{max}(\omega_3)$ represents the maximum number of paths of length 3 in the network, which is

$$\begin{aligned} N_{max}(\omega_3) = & \frac{n_3(n_3-1)(n_3-2)(n_3-2)}{2} \\ & - \frac{n_3(n_3-1)(n_3-2)}{3} \\ & + m_3(n_3-1)(n_3-2) + \frac{m_3(m_3-1)(2n_3-3)}{2}, \end{aligned} \tag{7}$$

where n_3 is the number of nodes that achieve full connectivity. m_3 represents the number of edges remaining after n_3 nodes achieve full connectivity, $m_3 = M - n_3(n_3-1)/2$.

After knowing the true contributions of all the paths that connect node pairs, the tendency to generate a link is represented by the sum of them. This is called as the path-based Bayesian (PB) method, which is given as

$$s_{xy}^{PB} = \sum_{k=1}^{L} \sum_{\omega_k \in O_{xy}} P'(A_1|\omega_k), \tag{8}$$

where s_{xy}^{PB} is the score of PB method. O_{xy} represents the set of all paths between node pair (x,y).

The way to calculate PB method is

I. We first divide the network adjacency matrix A into a training set A^T and a probe set A^R.
II. To calculate the contribution of a path of length k, we need to locate the core of this path, which is included in $(A^T)^{k-2}$.
III. When $k = 2$, the core of paths constitutes unit matrix I. Depending on the core of paths, we can calculate the prior contribution of each path of length 2, which is $C_2(i,i)$. When $k \geq 3$, for any node pairs $ij(i<j)$ in the upper triangular matrix, if there are cores between them, i.e., $(A^T)^{k-2} > 0$, we calculate its prior contribution, which is $C_k(i,j)$. Since the network is

undirected, only the upper triangular matrix need to be calculated, i.e., $C_k(j,i) = C_k(i,j)$.

IV. At the beginning, $N_L(i,j) = 1$, $N_U(i,j) = 1$. In order to calculate $C_k(i,j)$, the first step is to go through all the combination of node i's neighbors $N(i)$ and node j's neighbors $N(j)$ to get the prior contribution $C_k(i,j)$. When $A^T(N(i), N(j)) > 0$, we add one to $N_L(i,j)$. We add one to $N_U(i,j)$ in other cases. The prior contribution is calculated according to Eq. 2.

V. After getting prior contribution of each path, $P(\omega_k)$, and $E(P(\omega_k))$, the posterior contribution of a path of length k can be obtained by the Eq. 5.

VI. We modify the value of length k and repeat step II to step V until the posterior contribution of all the paths that needs to be obtained is calculated.

VII. We need to traverse all the the node pairs ij without links in the network and calculate the posterior probability of the link generation between them. Since the network is undirected, only the upper triangular matrix needs to be calculated. The score of node pairs is the sum of the posterior contribution of all the paths between node pairs, which can be calculated according to the Eq. 8.

5 Evaluation Method

Like most studies, this paper uses AUC and Precision to evaluate prediction results of link prediction methods [16]. When calculating AUC [17], we repeatedly compare the score of a random edge that should be predicted with that of a random edge that should not be predicted. If the score of the edge that should be predicted is higher, we add 1 node. When the scores of both edges are equal, 0.5 nodes will be added. We repeat this experiment independently n times. If there are n' times that edge should be predicted has a higher score, and n'' times that both edges have same score, AUC can be calculated as

$$AUC = \frac{n' + 0.5 * n''}{n}. \tag{9}$$

If there are t edges in both the probe set and top T ranked edges, Precision [18] can be represented as

$$AUC = \frac{t}{T}. \tag{10}$$

Both of these indexes estimate the accuracy of the link prediction algorithm by predicting the probe set, but they have different perspectives on the prediction algorithm. AUC measures the performance of the algorithm as a whole, while Precision only considers the prediction accuracy of T top-ranked edges. In a network, if one of the two indexes has the same score, then the link prediction method that the other index performs better is more suitable for this network.

In order to compare the prediction effects of the algorithm, this paper introduces four traditional indexes and an index similar to the proposed algorithm, and then compares their prediction effects with the proposed algorithm. Besides, all of these indexes are based on the topology of the network.

The Common Neighbor (CN) index is one of the most important link prediction algorithms. This algorithm believes that more common neighbors are more likely to facilitate links between node pairs, that is

$$s_{xy}^{CN} = |R(x) \cap R(y)|, \tag{11}$$

where $R(x)$ represents the set of neighbors of node x. $|R(x) \cap R(y)|$ represents the number of common neighbors of the node pairs xy.

The contribution of the common neighbors to Adamin-Adar (AA) index is related to the degree of the common neighbors. It is proportional to the reciprocal of the log of the common neighbor degree, which is

$$s_{xy}^{AA} = \sum_{z \in R(x) \cap R(y)} \frac{1}{logk_z}, \tag{12}$$

where $z \in R(x) \cap R(y)$ means node z in the set of common neighbors of the node pairs xy. k_z refers to the degree of node z.

The difference between the Resource Allocation (RA) index and the AA indicator is that the RA index considers that the resources passing through the nodes are equally allocated, so the contribution of the common neighbors is inversely proportional to the degree of the common neighbors, which is

$$s_{xy}^{RA} = \sum_{z \in R(x) \cap R(y)} \frac{1}{k_z}. \tag{13}$$

The Katz index counts paths of the same length between pairs of nodes. Meanwhile, it assigns different weights to reflect the difference in the contribution of paths of different length to the formation of links between node pairs, which is

$$s_{xy}^{Katz} = \sum_{l=1}^{\infty} \alpha^l |path_{xy}^l|, \tag{14}$$

where α is a hyperparameter. l indicates the length of paths. It affects the weights of paths of different lengths as well. Generally speaking, it is meaningless to calculate a path that is too long, because the information in the network is often time-sensitive and almost no information will be transmitted through a long path. Therefore, the path with a length of 3 or 4 is generally counted at most.

Local Naive Bayes (LNB) index applies a machine learning model to link prediction problem based on local information. According to the Naive Bayesian model, the algorithm considers that the contribution of the common neighbors to the link generation between node pairs is the product of the posterior probabilities generated by the common neighbors, which is

$$P(A_1|R(x) \cap R(y)) = \frac{P(A_1)}{P(R(x) \cap R(y))} \prod_{z \in R(x) \cap R(y)} P(z|A_1), \tag{15}$$

$$P(A_0|R(x)\cap R(y)) = \frac{P(A_0)}{P(R(x)\cap R(y))}\prod_{z\in R(x)\cap R(y)} P(z|A_0), \tag{16}$$

where $R(x)\cap R(y)$ represents all the common neighbors between node pairs xy, and z is one of the common neighbors. After normalization, we can get the final score s_{xy}^{LNB}, which is

$$s_{xy}^{LNB} = |R(x)\cap R(y)|log\frac{P(A_0)}{P(A_1)} + \sum_{z\in R(x)\cap R(y)} log\frac{P(A_1|z)}{P(A_0|z)}. \tag{17}$$

If we further optimize this link prediction index by combining local information indexes, we can obtain LNB-CN, LNB-AA, and LNB-RA indexes. The formulas are as follows:

$$s_{xy}^{LNB-CN} = |R(x)\cap R(y)|log\frac{P(A_0)}{P(A_1)} + \sum_{z\in R(x)\cap R(y)} log\frac{P(A_1|z)}{P(A_0|z)}, \tag{18}$$

$$s_{xy}^{LNB-AA} = \sum_{z\in R(x)\cap R(y)} \frac{1}{logk_z}(log\frac{P(A_0)}{P(A_1)} + log\frac{P(A_1|z)}{P(A_0|z)}), \tag{19}$$

$$s_{xy}^{LNB-RA} = \sum_{z\in R(x)\cap R(y)} \frac{1}{k_z}(log\frac{P(A_0)}{P(A_1)} + log\frac{P(A_1|z)}{P(A_0|z)}). \tag{20}$$

6 Experiments and Discussion

In order to test the prediction effect of the link prediction method in this paper, the prediction accuracy is tested in the food network (Florida [19], Everglades [20], StMarks [21]), the biological network (C. elegans [22]), the social network (email-Eu-core temporal network [23], Political blogs [24]), the protein-protein interaction network (Yeast [25]) and the power network (Power [26]). To simplify the problem, the experiment was conducted in the undirected and unweighted network. In the network, the weights of edges are ignored, and all

Table 1. Performance of different link prediction methods under Precision index.

Network	CN	AA	RA	Katz	LNB-CN	LNB-AA	LNB-RA	PB
Florida	0.070	0.075	0.072	0.288	0.088	0.089	0.088	**0.309**
C.elegans	0.100	0.102	0.103	0.123	0.107	0.108	0.101	**0.135**
Everglades	0.153	0.163	0.173	0.374	0.169	0.185	0.192	**0.400**
StMarks	0.131	0.148	0.157	0.224	0.177	0.174	0.171	**0.257**
email-Eu-core-temporal	0.198	0.225	0.259	0.168	0.218	0.248	**0.272**	0.221
Political blogs	0.166	0.176	0.146	0.183	0.168	0.165	0.161	**0.188**
Yeast	0.149	0.179	0.254	0.250	0.151	0.189	0.280	**0.341**
Power	0.045	0.024	0.021	**0.045**	0.042	0.025	0.028	0.040

edges are considered to be bidirectional. In addition, the existence of a self-loop is not allowed in the network. The experimental results are shown in the following table.

Table 1 shows the prediction results of different link prediction methods under the Precision index. The method we studied is the PB method in the rightmost column of the table. The bold font represents the link prediction method with the highest prediction accuracy in the network. From Table 1, we can see that PB method has a good prediction effect under Precision index. Only in the Email and Power networks, the prediction effect of PB method is not the best. The reason is that single layer network can not provide enough effective information in extreme cases. For example, in Power network, since the network is too sparse, the existing probability of different length paths is very low. PB method judges the relative weights of different length paths according to the probability of path occurrence. The relative weights of different length paths are misjudged, which results in the wrong estimation of the score between nodes and the decrease of the prediction accuracy of the PB method. On the other hand, the excellent performance of PB method in other networks shows that PB method can more accurately describe the relationship between node pairs on the premise that the network can provide enough information, making the node pairs that tend to generate connections rank ahead.

Table 2. Performance of different link prediction methods under AUC index.

Network	CN	AA	RA	Katz	LNB-CN	LNB-AA	LNB-RA	PB
Florida	0.610	0.612	0.614	0.811	0.692	0.696	0.697	**0.849**
C.elegans	0.846	0.861	0.862	0.845	0.856	0.862	0.862	**0.890**
Everglades	0.693	0.698	0.712	0.838	0.731	0.735	0.734	**0.877**
StMarks	0.658	0.670	0.679	0.717	0.721	0.723	0.715	**0.781**
email-Eu-core-temporal	0.943	0.946	0.949	0.925	0.945	0.948	**0.950**	0.947
Political blogs	0.919	0.919	0.920	0.930	0.919	0.920	0.922	**0.941**
Yeast	0.895	0.894	0.892	0.930	0.888	0.892	0.898	**0.939**
Power	0.586	0.586	0.584	**0.628**	0.587	0.592	0.585	0.604

Table 2 shows the prediction effect of different link prediction methods under AUC index. AUC calculates the mean value of prediction effect under different thresholds, which reflects the overall prediction effect of link prediction method. As we can see from Table 2, PB method has the highest prediction accuracy in Florida network, C. elegans network, Everglades network, StMarks network, Blogs network and Yeast network. This phenomenon shows that PB method can calculate the score of node pairs more reasonably, and ensure that the node pairs that should be predicted score higher than those that should not be predicted.

Tables 1 and 2 show that PB method in this paper has high prediction accuracy under Precision and AUC indices in single layer networks. The reason is that this method not only quantifies the contribution of the path by using the clustering coefficient of the path, but also modifies the contribution of the path by using the global topological information. Our method integrates local topological information, global topological information and network evolution trend to predict links in the network which makes full use of topological information. The prediction effect is obviously better than that of traditional and similar indicators. In addition, for the reason that this algorithm does not need to determine the hyperparameter through repeated experiments, it effectively reduces the redundant experimental process.

Compared with the LNB index, the PB method is more reasonable. Although both of them believe that the probability of link generation between nodes is related to the paths between nodes, the LNB index treats the paths which connect node pairs as a kind of restriction. This idea holds true in image problems, because the correlation between pixels is very strong, but it is not always the case in the network. In LNB index, contributions of paths are multiplied as the contribution to the formation of links. On the contrary, PB method is convinced that the sum of the contribution of each path between node pairs represents the possibility to generate a link, which is more in line with common sense. Moreover, the final score generated by LNB index is related to the number of links existing in the network, that is, the more links there are in the network, the easier it is to generate links. However, in real life, when the network tends to be saturated, its desire to generate links should be extremely low. Our PB algorithm assigns weights of paths according to the actual network topology. The weights of paths of different lengths is proportional to its importance in the network, so the weight of each path is arranged more reasonably in the proposed algorithm.

7 Conclusion

In summary, this paper proposes a path-based Bayesian method for link prediction. In this method, we define clustering coefficient of paths which quantifies the priori contribution of paths to link generation. Then, we use Bayesian method to transform the priori contribution to the posterior contribution which reflects the true contribution of the path. The path-based Bayesian algorithm performs well compared to existing link prediction methods. Compared with the link prediction methods based on local information, it applies more network information and achieves better prediction accuracy. Besides, it reduces redundant calculations which is used by traditional path-based link prediction algorithm to determine relative weights of paths of different lengths. Furthermore, this method only makes simple use of global topological information which gives it better robustness. Last but not least, our method only uses topological information, so the information source is more reliable.

Although this method has many advantages, it still has room for further study. The use of extra information which can reflect the evolution trend of the network can make the weights of paths of different length more accurate. In the future, we are looking forward to improve this approach in more informative networks, such as multilayer networks.

Acknowledgements. This work was supported in part by National Natural Science Foundation of China (No. 61872187, No. 61871444, No. 61773215) and Major Special Project of Core Electronic Devices, High-end Generic Chips and Basic Software (No. 2015ZX01041101).

References

1. Chen, S., Huang, W., Cattani, C., Altieri, G.: Traffic dynamics on complex networks: a survey. Math. Probl. Eng. **2012** (2012)
2. Taskar, B., Wong, M.-F., Abbeel, P., Koller, D.: Link prediction in relational data. In: Advances in Neural Information Processing Systems, pp. 659–666 (2004)
3. Barzel, B., Barabási, A.-L.: Network link prediction by global silencing of indirect correlations. Nat. Biotechnol. **31**(8), 720 (2013)
4. Lü, L., Zhou, T.: Link prediction in complex networks: a survey. Phys. A **390**(6), 1150–1170 (2011)
5. Lorrain, F., White, H.C.: Structural equivalence of individuals in social networks. J. Math. Sociol. **1**(1), 49–80 (1971)
6. Adamic, L.A., Adar, E.: Friends and neighbors on the web. Soc. Netw. **25**(3), 211–230 (2003)
7. Zhou, T., Lü, L., Zhang, Y.-C.: Predicting missing links via local information. Eur. Phys. J. B **71**(4), 623–630 (2009)
8. Katz, L.: A new status index derived from sociometric analysis. Psychometrika **18**(1), 39–43 (1953)
9. Lü, L., Pan, L., Zhou, T., Zhang, Y.-C., Stanley, H.E.: Toward link predictability of complex networks. Proc. Natl. Acad. Sci. **112**(8), 2325–2330 (2015)
10. Miller, K., Jordan, M.I., Griffiths, T.L.: Nonparametric latent feature models for link prediction. In: Advances in Neural Information Processing Systems, pp. 1276–1284 (2009)
11. Sarukkai, R.R.: Link prediction and path analysis using Markov chains. Comput. Netw. **33**(1–6), 377–386 (2000)
12. Al Hasan, M., Chaoji, V., Salem, S., Zaki, M.: Link prediction using supervised learning. In: SDM06: Workshop on Link Analysis, Counter-Terrorism and Security (2006)
13. Liu, Z., Zhang, Q.-M., Lü, L., Zhou, T.: Link prediction in complex networks: a local naïve Bayes model. EPL (Eur. Lett.) **96**(4), 48007 (2011)
14. Wang, H., Shi, X., Yeung, D.-Y.: Relational deep learning: a deep latent variable model for link prediction. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
15. Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K., Kertesz, J.: Generalizations of the clustering coefficient to weighted complex networks. Phys. Rev. E **75**(2), 027105 (2007)
16. Yang, Y., Lichtenwalter, R.N., Chawla, N.V.: Evaluating link prediction methods. Knowl. Inf. Syst. **45**(3), 751–782 (2015)
17. Hanley, J.A., McNeil, B.J.: The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology **143**(1), 29–36 (1982)
18. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. (TOIS) **22**(1), 5–53 (2004)

19. Ulanowicz, R.E., Bondavalli, C., Egnotovich, M.S.: Network analysis of trophic dynamics in South Florida ecosystem, FY 97: the Florida Bay ecosystem. Annual Report to the United States Geological Service Biological Resources Division Ref. No. [UMCES] CBL, pp. 98–123 (1998)
20. Ulanowicz, R.E., Bondavalli, C., Heymans, J.J., Egnotovich, M.S.: Network analysis of trophic dynamics in South Florida ecosystem, FY 99: the graminoid ecosystem. Annual Report to the United States Geological Service Biological Resources Division Ref. No. [UMCES] CBL 00–0176, Chesapeake Biological Laboratory, University of Maryland (2000)
21. Baird, D., Luczkovich, J., Christian, R.R.: Assessment of spatial and temporal variability in ecosystem attributes of the St Marks national wildlife refuge, Apalachee Bay, Florida. Estuar. Coast. Shelf Sci. **47**(3), 329–349 (1998)
22. Stiernagle, T.: Maintenance of c. elegans. C. elegans **2**, 51–67 (1999)
23. Paranjape, A., Benson, A.R., Leskovec, J.: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017 (2017)
24. Ackland, R., et al.: Mapping the us political blogosphere: are conservative bloggers more prominent? In: BlogTalk Downunder 2005 Conference, Sydney (2005)
25. Von Mering, C., et al.: Comparative assessment of large-scale data sets of protein-protein interactions. Nature **417**(6887), 399 (2002)
26. Watts, D.J., et al.: Nature (London) **393**, 440 (1998)