

Predicting missing links via correlation between nodes

Hao Liao^{a,b}, An Zeng^{a,d,*}, Yi-Cheng Zhang^{c,e}

^a Alibaba Research Center for Complexity Sciences, Alibaba Business College, Hangzhou Normal University, Hangzhou 311121, PR China

^b Guangdong Province Key Laboratory of Popular High Performance Computers, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, PR China

^c Department of Physics, University of Fribourg, Chemin du Musée 3, CH-1700 Fribourg, Switzerland

^d School of Systems Science, Beijing Normal University, Beijing 100875, PR China

^e School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, PR China

As a fundamental problem in many different fields, link prediction aims to estimate the likelihood of an existing link between two nodes based on the observed information. Since this problem is related to many applications ranging from uncovering missing data to predicting the evolution of networks, link prediction has been intensively investigated recently and many methods have been proposed so far. The essential challenge of link prediction is to estimate the similarity between nodes. Most of the existing methods are based on the common neighbor index and its variants. In this paper, we propose to calculate the similarity between nodes by the Pearson correlation coefficient. This method is found to be very effective when applied to calculate similarity based on high order paths. We finally fuse the correlation-based method with the resource allocation method, and find that the combined method can substantially outperform the existing methods, especially in sparse networks.

1. Introduction

The ultimate objective of many scientific studies is to do prediction. For instance, understanding the mechanism of epidemic spreading can help us to predict the future coverage of a certain virus [1], the mechanistic model for the citation dynamics of individual papers can be applied to predict the future evolution of scientific publications [2]. While mathematical models and prediction techniques are sufficiently mature for some systems, reliable prediction approaches are still unavailable in most systems. Besides the prediction of the collective behavior, the prediction in microscopic level, such as the well-known link prediction challenge in complex networks, has also attracted a lot of attention.

* Corresponding author at: School of Systems Science, Beijing Normal University, Beijing 100875, PR China.
E-mail address: anzeng@bnu.edu.cn (A. Zeng).

Link prediction is a very important problem that aims at estimating the likelihood of the existence of a link between two nodes [3]. Solving this problem cannot only help us complete the missing data in biological networks such as the protein-protein interaction networks and metabolic networks [4,5], but also enable us to predict the evolution of social networks [6,7]. In fact, link prediction is also closely connected to some other problems such as recommendation [8] and spurious links detection [9]. A sound link prediction method will help to design more efficient recommendation algorithm to filter out irrelevant information for online users [10]. Moreover, the link prediction method can also be applied to analyzing the reliability of existing links and accordingly identifying some noisy connections of networks. The progress in this field will largely push forward the research in other fields. Accordingly, the problem of missing link prediction has been intensively studied by researchers from different backgrounds and many methods applied to different fields have been proposed [11–14]. For a review, see Ref. [15].

The basic assumption for link prediction is that two nodes are more likely to have a link if they are similar to each other. Therefore, the essential problem for link prediction is how to calculate the similarity between nodes accurately. One of the most straightforward methods is called common neighbor which measures the similarity between two individuals by directly counting the number of common neighboring nodes [16]. However, this method has serious shortcomings as it strongly favors the large degree nodes. To solve this problem, many variants, such as the Jaccard index [17] and Salton index [18], have been applied to remove this tendency. In addition, some other methods including Katz index [19], sim-rank [20], hierarchical random graph [5] and stochastic block model [21,22] are also very effective in estimating nodes' similarity. However, these methods are based on global algorithms that can be prohibitive to use for large-scale systems.

In this paper, we argue that the similarity between nodes can be calculated based on another completely different type of method, namely correlation coefficient. In broad definition, correlation refers to any class of statistical relationships involving dependence between two or more random variables. In our case, it actually refers to the Pearson correlation [23] between nodes' attribute vectors which can come from the adjacency matrix or higher order of that. In link prediction, one of the biggest challenges is the data sparsity. It means that a lot of data at hand is too sparse to extract valuable similarity information from the simple common neighbor method or its variants. One possible solution has been discussed in Ref. [24] in which longer paths (i.e. paths with length larger than 2) are applied to measure nodes' similarity. However, when it comes to such high order information, much noise will be included so that the similarity matrix is indeed denser but the similarities are not satisfactorily accurate, which leads to a poor outcome of predicted links. In our simulation, we find that the correlation-based method is very effective when applied to calculating similarity based on high order paths. We finally use the new method with the resource allocation method [25], and find that the combined method can substantially outperform the existing methods, especially in sparse networks.

2. Related works

To begin our analysis, we first briefly describe the link prediction problem and review some representative methods. Considering an unweighted undirected simple network $G(V, E)$, V is the set of nodes and E is the set of links. The multiple links and self-connections are not allowed. For each pair of nodes x, y belonging to V , we calculate a score s_{xy} which measures the likelihood for nodes x and y to have a link between them. Since G is undirected, the score is supposed to be symmetry, i.e. $s_{xy} = s_{yx}$. All the nonexistent links are sorted in a decreasing order according to their s scores, and the links on the top are more likely to exist. There are many different ways to calculate s_{xy} score and the most common and straightforward way is to calculate the similarity between nodes x and y .

Generally speaking, two nodes are considered to be similar if they have some common important features in topology [15]. In this paper, we compare the prediction accuracies of four typical similarity indices: Common neighbor (CN), Resource allocation (RA), Jaccard and Local path. Their definitions and relevant motivations are introduced as follows:

- (i) *Common Neighbor* (CN). Two nodes x and y are more likely to form a link if they have many common neighbors. Let $\Gamma(x)$ denote the set of neighbors of node x , the simplest measure of the neighborhood overlap can be directly calculated as:

$$s_{xy}^{\text{CN}} = |\Gamma(x) \cap \Gamma(y)| \quad (1)$$

which is the actual aggregation method used by most websites. However, the drawback of CN is that it is in favor of the nodes with large degree. It is obvious that $s_{xy} = (A^2)_{xy}$, where A is the adjacency matrix. $A_{xy} = 1$ if x and y are directly connected and $A_{xy} = 0$ otherwise. Note that $(A^2)_{xy}$ is also the number of different paths with length 2 connecting x and y . Newman [15] used this quantity in the study of collaboration networks, showing the correlation between the number of common neighbors and the probability that two scientists will collaborate in the future. Therefore, we here select CN as the representative of all CN-based measures. Although CN consumes little time and performs relatively good among many local indices, due to the insufficient information, its accuracy cannot catch up with the measures based on global information. One typical example is the Katz index [19].

- (ii) *Jaccard coefficient* (Jaccard). This index was proposed by Jaccard over a hundred years ago. The algorithm is a traditional similarity measurement in the literature. It is defined as

$$s_{xy}^{\text{Jaccard}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}. \quad (2)$$

The motivation of this index is that the pure common neighbor favors a lot the large degree nodes: it is easier for large degree nodes to form common neighbors with other nodes. The denominator can remove the tendency for high degree nodes to have high similarity with other nodes. Note that, there are many other ways to remove the tendency of CN to large degree nodes, such as cosine index, Sorensen index, Hub promoted index and so on [15].

- (iii) *Resource allocation* (RA). This index is inspired by the resource allocation dynamics on complex network. Considering a pair of nodes, x and y , which are not connected directly, suppose that the node x needs to give some resource to y , with their common neighbors as transmitters. Each transmitter has a single unit of resource and will distribute it to all its neighbors equally. The similarity between x and y can be directly calculated as the amount of resource y received from x :

$$S_{xy}^{\text{RA}} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{k_z}. \quad (3)$$

This measure is symmetric as well. In fact, there is a similar similarity index called Adamic-Adar (AA) Index. Instead of using the degree of the common neighbor z (k_z), $\log(k_z)$ is used to depress the contribution of the high-degree common neighbors. The difference between RA and AA is similar if k_z is small. However, in heterogeneous networks where k_z can be very large, the difference of RA and AA becomes large. By punishing the high degree common neighbors heavily, RA usually achieves a higher link prediction accuracy than AA.

- (iv) *Local Path index* (LP). This index was introduced in Ref. [24]. This index takes local paths into consideration, with wider horizon than CN. It is given by

$$S_{xy}^{\text{LP}} = A^2 + \epsilon * A^3 \quad (4)$$

where ϵ is chosen as a free parameter and A is the adjacency matrix of the network. Note that LP degenerates to CN when $\epsilon = 0$. Moreover, this similarity index can be regarded as a simplified Katz index [19]. Instead of taking all paths into account in the network, LP only considers the paths with length three. It has been shown that LP can achieve similar performance to Katz, but with a lower computational complexity.

3. Similarity based on correlation between nodes

The above methods, though effective in link prediction, all measure the similarity between nodes based on the common neighbor information. In this paper, we propose to calculate node similarities by the correlation between nodes. In fact, similar idea has been applied to design ranking algorithm for online users' reputation [26]. Given a vector \mathbf{v}_x (\mathbf{v}_y) describing the feature of a node x (y), we calculate the similarity between these two nodes based on the Pearson correlation coefficient of v_x and v_y . Mathematically, it reads

$$S_{xy}^{\text{Corr}} = \frac{1}{N} \sum_{l=1}^N \frac{(v_{xl} - \bar{v}_x)(v_{yl} - \bar{v}_y)}{\sigma_{v_x} \sigma_{v_y}} \quad (5)$$

where \bar{v}_x and σ_{v_x} are respectively the mean and standard deviation of vector \mathbf{v}_x . As discussed above, \mathbf{v}_x should be an attribute vector for node x . One simple way would be to directly set it as $v_{xl} = A_{xl}$. In this paper, we go beyond the adjacency matrix and take A^m into consideration (m can be larger than 1), so that we set \mathbf{v}_x as the corresponding column of the A^m .

4. Data and metrics

To evaluate the effectiveness of the above methods, we consider nine empirical networks including both social networks and nonsocial networks: (i) Dolphin: a dolphin friendship network, which is an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand [27]. (ii) Jazz: a music collaboration network obtained from The Red Hot Jazz Archive digital database. Here it includes 198 bands that performed between 1912 and 1940, with most of the bands in the 1920 to 1940 [28]. (iii) C.elegans: the neural network of the nematode worm *C.elegans*, in which edge joins two neurons if they are connected, by either a synapse or a gap junction [29]. (iv) USAIR: the US air transportation network [30], which contains 332 airports and 2126 airlines. (v) Netscience: a coauthorship network between scientists who are publishing on the topic of network science [31]. This network contains 1589 scientists, and 128 of whom are isolated. Moreover, it is consisted of 268 connected components, and the size of the largest connected component is only 379. (vi) Email: an email communication network [32]. (vii) TAP: a yeast protein binding network generated by tandem affinity purification experiments [33]. (viii) Power Grid: the electrical power grid of western US [29], with nodes representing generators, transformers and substations, and links corresponding to the high-voltage transmission lines between them. This network contains 4941 nodes, and they are well connected. (ix) HEP: a collaboration network of high energy physicists, which contains the collaboration network of scientists posting preprints on the high-energy theory archive at www.arxiv.org from 1995 to 1999 [31]. We only take into account of the giant component of these networks. This is because for a pair of nodes located in two disconnected components, their S_{xy} score will be zero according to CN and its variants. Table 1 shows the basic statistics of all the giant components of those networks.

Table 1

Structural properties of the different real networks. Structural properties include network size (N), edge number (E), degree Heterogeneity ($H = \langle k^2 \rangle / \langle k \rangle^2$), degree assortativity (r), clustering coefficient ($\langle C \rangle$) and average shortest path length ($\langle d \rangle$).

Network	N	E	H	r	$\langle C \rangle$	$\langle d \rangle$
Dolphins	62	159	1.327	-0.044	0.259	3.357
Jazz	198	2,742	1.395	0.020	0.618	2.235
C.elegans	297	2,148	1.801	-0.163	0.292	2.455
USAIR	332	2,126	3.464	-0.208	0.749	2.46
Netscience	379	914	1.663	-0.082	0.741	6.042
Email	1133	5,451	1.942	0.078	0.220	3.606
TAP	1373	6,833	1.644	0.579	0.529	5.224
PowerGrid	4941	6,594	1.450	0.004	0.080	18.989
HEP	5835	13,815	1.926	0.185	0.506	7.026

Table 2

Comparison of different algorithms' accuracy quantified by AUC in each real network. The training set contains 90% of the known links. Each number is obtained by averaging over 10 implementations with independently random divisions of training set and probe set. We set the parameters $\epsilon = 10^{-3}$ in LP and $\epsilon = 10^{-2}$ in HCR. The highest accuracy in each line is emphasized by boldface.

Network	CN	RA	Jaccard	LP	HCR
Dolphins	0.803	0.806	0.802	0.829	0.846
Jazz	0.955	0.971	0.958	0.947	0.973
C. elegans	0.846	0.867	0.811	0.866	0.881
USAIR	0.954	0.972	0.915	0.952	0.974
Netsci	0.981	0.985	0.981	0.989	0.992
Email	0.858	0.859	0.856	0.919	0.922
TAP	0.954	0.954	0.956	0.967	0.977
PowerGrid	0.624	0.624	0.625	0.689	0.767
HEP	0.941	0.943	0.942	0.961	0.965

For each of the real network, the observed links M are randomly divided into two parts: the training set M^T , treated as known information, while probe set M^P is used for verifying the prediction accuracy and no information in that is allowed to be used for prediction. M^T plus M^P is the whole data set. Note that, each time before moving a link to the probe set we first check if this removal will make the training network disconnected. Usually, the training set contains 90% of the links and the probe set consists of 10% links. In this paper, we employ a standard metric, area under the receiver operating characteristic curve (AUC) [34] to measure the accuracy of the prediction. AUC can be interpreted as the probability that a randomly chosen missing link from M^P is given a higher score than a randomly chosen nonexistent link. In practice, we usually calculate the score of each non-observed link instead of giving the ordered list since latter task is more time-consuming. Then, AUC requires n times of independent comparisons. At each time step, we randomly choose a missing link and a nonexistent link to compare their scores. After the comparison, we record there are n_1 times the missing link having a higher score, and n_2 times they have the same score. The final AUC is calculated as $AUC = (n_1 + 0.5 \times n_2) / n$. If all the scores are given by an independent and identical distribution, then AUC should be around 0.5. A higher AUC is corresponding to a more accurate prediction.

5. Results

We first compare the performance of our method in four representative data sets: Jazz, Netscience, Email and TAP. The detailed AUC values on other data sets will be reported in Table 2. In fact, we observe in our simulation that s_{xy}^{Corr} itself cannot outperform the traditional similarity measure such as CN, Jaccard, RA and LP in link prediction. However, we find that s_{xy}^{Corr} works well in extracting the node similarity information from high order paths (i.e. paths with length larger than 2). In order to show this, we set $m = 2$ in s_{xy}^{Corr} and compare it with $s = A^3$ which directly uses the number of paths with length 3 to measure the similarity between nodes. Note that the A^3 method has already been applied to combining with the common neighbor index in the LP method to solve the data sparsity problem.

We investigate the effect of data sparsity on the s_{xy}^{Corr} and A^3 methods. To this end, we move fraction p of all links to the probe set and use the remaining fraction $1 - p$ of all links as the training set. A larger p is corresponding to a more sparser known information of the real network. The AUC results of both methods under different p are presented in Fig. 1. One can see that in all networks, AUC of the s_{xy}^{Corr} is significantly higher than that of A^3 . Interestingly, the advantage of s_{xy}^{Corr} to A^3 becomes generally larger when the fraction of links in the training set is smaller. These results are actually very important since the high order paths are usually applied to solve the data sparsity problem. Generally speaking, the data sparsity problem could be solved more effectively if the high order paths information is used properly.

Inspired by the results above, we propose to combine the s_{xy}^{Corr} method and one of the traditional similarity methods to achieve higher accuracy in link prediction. As the RA method is one of the most efficient ones among the variants of the CN

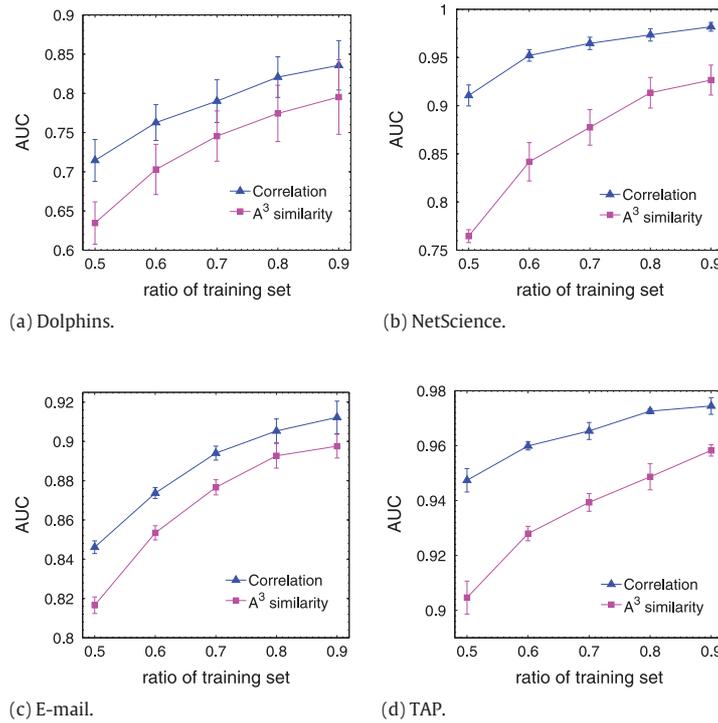


Fig. 1. (color online) AUC of the s_{xy}^{Corr} and A^3 methods as a function of $1 - p$ (the fraction of links in the training set) in four real networks. The error bars are obtained based on 10 independent realizations.

method, we adopt it to design a new method. As the new method is a Hybrid of the Correlation method and the Resource allocation method, we refer it as the HCR method in this paper. The formula of the HCR method reads

$$s_{xy}^{\text{HCR}} = s_{xy}^{\text{RA}} + \epsilon * s_{xy}^{\text{Corr}}, \quad (6)$$

where ϵ is a tunable parameter. Here, we set $m = 2$ for s_{xy}^{Corr} . In fact, s_{xy}^{RA} already enjoys a high prediction accuracy in dense data and s_{xy}^{Corr} can accurately predict missing links with very sparse information. Therefore, the HCR method is a very general link prediction method which is supposed to work well both in dense and sparse networks.

To validate the HCR method, we study the dependence of its AUC on ϵ in four real networks in Fig. 2. One can see that the prediction results are substantially improved once ϵ is larger than 0, which indicates that the s_{xy}^{Corr} method can indeed improve the s_{xy}^{RA} method (corresponding to ϵ). We also compute the results for the rest five real networks. Consistent with Fig. 2, we observe that the AUC jumps to a bigger value as long as ϵ is larger than 0. This is because a lot of node pairs in these sparse networks have zero resource allocation similarity. After introducing the correlation method, the tie is broken and these node pairs become distinguishable (the nodes with links in the probe set can be ranked higher than the others). Consequently, the AUC is improved. This mechanism is actually very similar to LP where the high order similarity (A^3) is used to improve the common neighbor similarity (A^2). After the tie is broken, further enlarging ϵ will not lead to additional improvement in AUC. As the LP method was proposed to solve the data sparsity problem as well, we compare the HCR method with it in Fig. 2. One can see that, when $\epsilon > 0$ HCR method can outperform the LP method, indicating the data sparsity problem is better addressed in the HCR method. This result is actually reasonable, as we already observe above that s_{xy}^{Corr} can outperform A^3 in sparse networks.

The results of the real networks are also reported in Table 2. One can see as well that the HCR method generally has higher AUC than other methods in all the networks considered. Among these networks, power grid is a very sparse one. The similarity indices based on local information, such as the CN, RA and Jaccard, are all with low AUC. In other words, different normalization to CN in this case cannot make much difference. However, once the semi-local information is taken into account, the LP method can significantly improve the AUC by nearly 11%. Interestingly, the HCR method performs even better than LP and improve the AUC by more than 23%. Similar phenomenon can be seen in Email network as well. On the other hand, when the network is dense, such as the Jazz and USAir network, the LP method cannot outperform the CN method as the information from high order paths in this case is too noisy. In contrast, the HCR method has higher AUC than CN and other method, showing that HCR can make better use of the high order paths information than LP.

In link prediction, it is generally difficult to predict the missing link of the nodes with small degree. This is known as the ‘‘cold-start’’ problem [15]. In the literature, it has already been shown that the item cold-start problem can be well addressed

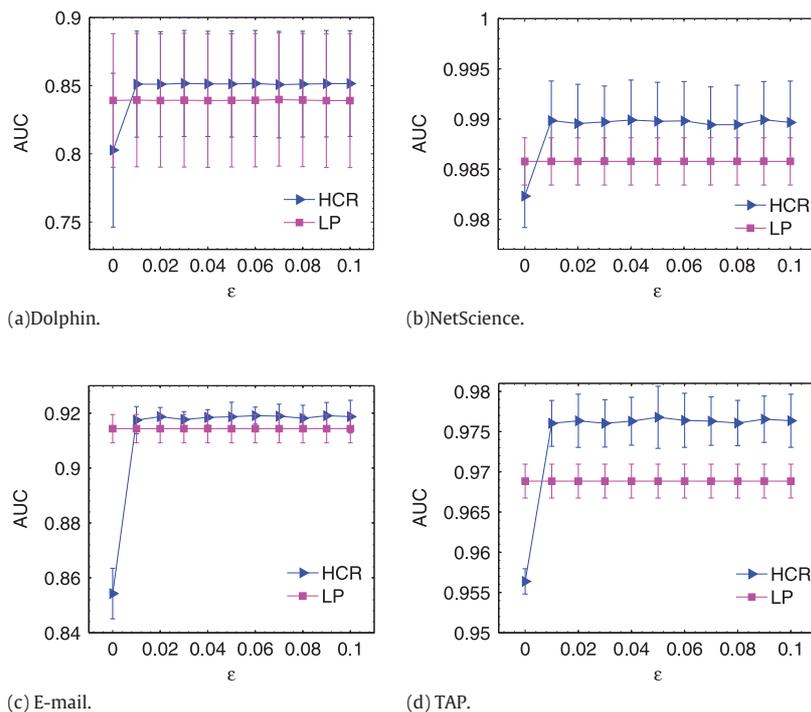


Fig. 2. (color online) The dependence of the AUC of the HCR method on ϵ in four real networks. The results of the LP method are shown for comparison. In the LP method, the parameter is chosen as $\epsilon = 0.01$ which is shown to be the optimal parameter for this method according to Ref. [24]. The error bars in this figure are obtained based on 10 independent realizations.

by changing the denominator in the CN method [35]. More specifically, the prediction accuracy for small degree nodes can be largely improved when larger score is given to the node pairs with small degree. However, in sparse networks the cold-start problem cannot be effectively solved in this way. More specifically, the AUC cannot be substantially increased by just changing the denominator in the CN method.

In fact, the essential difficulty for the cold-start problem is that the available information for the small degree nodes is too limited for the algorithms to accurately predict their missing links. The LP and HCR can address the cold-start problem by incorporating more information from high order paths. In order to show this, we pick the nodes with degree smaller than k , and report the prediction accuracy (AUC) of the probe set links between them in Fig. 3. We compare CN, LP and HCR methods in Fig. 3. As expected, one can see that AUC generally increases with k , indicating that the links connecting small degree nodes are indeed more difficult to be correctly predicted. Moreover, it is clear that the LP method can indeed result in a higher AUC than CN. The HCR method can significantly improve the AUC of small degree nodes. Therefore, we conclude that HCR is more effective in solving the cold-start problem than the LP method. In Fig. 3, HCR consistently outperforms LP and CN in all the four real networks we considered. The difference between HCR and the other two methods is more significant in the Dolphin and Email networks.

In principle, one can extend the current HCR method to deal with even higher order paths, and the modified HCR method reads

$$s_{xy}^{\text{HCR}} = s_{xy}^{\text{RA}} + \epsilon \sum_m s_{xy}^{\text{Corr}}(m). \quad (7)$$

The simulation results show that AUC can be slightly improved with higher order paths. We also consider an extension of the LP method,

$$s_{xy}^{\text{LP}} = A^2 + \epsilon \sum_m A^m. \quad (8)$$

However, the AUC of LP decreases when $m > 3$. These results evidently support again that HCR is more effective than LP in extracting similarity information, especially when the original information contains noise.

6. Conclusion and discussion

In this paper, we employed the Pearson correlation coefficient to measure the similarity between nodes, and accordingly apply it to predict the future links. We found that though the correlation method cannot outperform the common neighbor

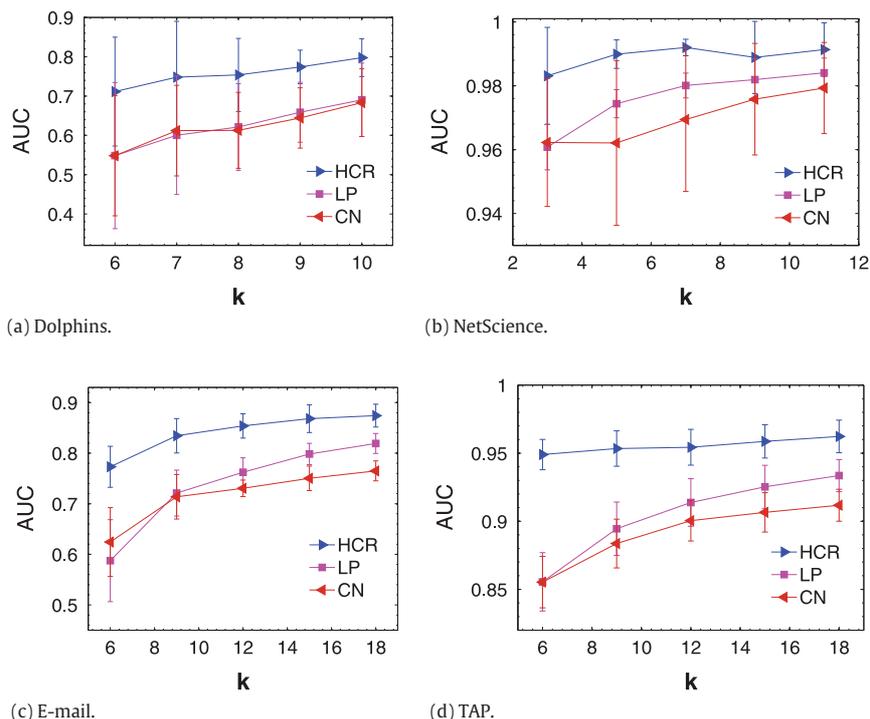


Fig. 3. (color online) The AUC of the probe set links connecting nodes with degree sum smaller than k when different link prediction algorithms are applied. In the LP method, the parameter is chosen as $\epsilon = 0.01$. In the HCR method, the parameter is chosen as $\epsilon = 0.001$. The error bars are obtained based on 10 independent realizations.

and its variants in link prediction, the correlation method is actually very efficient in extracting the similarity information from the high order path information. This is because the Pearson correlation coefficient is generally more robust to noise than the traditional indices based on common neighbor. We further combined the correlation method and the resource allocation method, and found that this hybridization can outperform the existing link prediction methods, especially when the available information from the observed network is little. We compared the new method with one existing method that is intended to solve the data sparsity problem, and the results showed that our method has higher accuracy.

Many issues remain still open. Our work implies that the Pearson correlation coefficient is more resistant to noisy information than the other methods. An interesting extension would be to investigate the link prediction problem in the noisy environment, i.e. the observed network containing some noisy links. One can compare the correlation-based method and the other methods, and systematically study the robustness of these methods to noise. The Pearson correlation opens a new direction for measuring the similarity between nodes. In fact, there are some other correlation coefficients such as Spearman [36] and Kendall's tau [37] coefficient. A detailed study of their performance in link prediction would be another interesting extension.

Our results also show that the high order paths in networks also contain some valuable information to characterize node similarity. This information is especially important for sparse networks. Similar study has already been conducted in recommender systems where the semi-local diffusion is found to be able to significantly improve the recommendation accuracy [38]. However, if such information is not used properly, too much noise will be involved and may jeopardize the predict accuracy [39]. Therefore, the link prediction method that is tolerant of noise is very important. In this paper, we present a possible method to solve this problem. There are some other possible ways for this problem, such as only taking into account the salient high order paths. Related methods ask for investigation in the future.

Acknowledgments

This work was partially supported by the EU FP7 Grant 611272 (project GROWTHCOM) and by the Swiss National Science Foundation (grant no. 200020-143272). The authors would like to acknowledge the support from the Opening Foundation of Alibaba Research Center for Complex Sciences, Hangzhou Normal University under Grant No. PD12001003002008 and PD12001003002006, H. L acknowledges support from National Science Foundation of China under Grant Nos. U1301252 and 61170076, Guangdong Natural Science Foundation under Grant No. 2014A030313553, and Fundamental Research Funds for the Central Universities under Grant No. 2014ZM0079.

References

- [1] D. Brockmann, D. Helbing, The hidden geometry of complex, network-driven contagion phenomena, *Science* 342 (2013) 1337.
- [2] D. Wang, C. Song, A.L. Barabasi, Quantifying long-term scientific impact, *Science* 342 (2013) 6154.
- [3] D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks, *J. Assoc. Inf. Sci. Technol.* 58 (2007) 7.
- [4] S. Redner, Teasing out the missing links, *Nature* 453 (2008) 47.
- [5] A. Clauset, C. Moore, M.E.J. Newman, Hierarchical structure and the prediction of missing links in networks, *Nature* 453 (2008) 98.
- [6] L.A. Adamic, E. Adar, Friends and neighbors on the web, *Social Networks* 25 (2003) 211.
- [7] D. Wang, D. Pedreschi, C. Song, F. Giannotti, A.L. Barabasi, Human mobility, social ties, and link prediction, in: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011.
- [8] Z.K. Zhang, C. Liu, Y.C. Zhang, T. Zhou, Solving the cold-start problem in recommender systems with social tags, *Europhys. Lett.* 92 (2010) 2.
- [9] A. Zeng, G. Cimini, Removing spurious interactions in complex networks, *Phys. Rev. E* 85 (2012) 036101.
- [10] J. Zhang, X. Kong, P.S. Yu, Predicting social links for new users across aligned heterogeneous social networks, in: *Proceedings of IEEE International Conference on Data Mining*, 2013.
- [11] R.N. Lichtenwalter, J.T. Lussier, N.V. Chawla, New perspectives and methods in link prediction, in: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.
- [12] Z. Lu, B. Savas, W. Tang, I. Dhillon, Supervised link prediction using multiple sources, in: *Proceedings of IEEE International Conference on Data Mining*, 2010.
- [13] D. Shin, S. Si, I.S. Dhillon, Multi-scale link prediction, in: *Proceedings of ACM international conference on Information and knowledge management*, 2012.
- [14] L. Lu, T. Zhou, Link prediction in weighted networks: The role of weak ties, *Europhys. Lett.* 89 (2009) 18001.
- [15] L. Lu, T. Zhou, Link prediction in complex networks: A survey, *Physica A* 390 (2011) 1150.
- [16] F. Lorrain, H.C. White, Structural equivalence of individuals in social networks, *J. Math. Sociol.* 1 (1971) 49.
- [17] P. Jaccard, Étude comparative de la distribution florale dans une portion des Alpes et des Jura, *Bull. Soc. Vaudoise Sci. Nat.* 37 (1901) 142.
- [18] G. Salton, A. Wong, S.C. Yang, A vector space model for automatic indexing, *Commun. ACM* 18 (1975) 613.
- [19] L. Katz, A new status index derived from sociometric analysis, *Psychometrika* 18 (1953) 1.
- [20] G. Jeh, J. Widom, SimRank: A measure of structural-context similarity, in: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.
- [21] B. Karrer, M.E.J. Newman, Stochastic blockmodels and community structure in networks, *Phys. Rev. E* 83 (2011) 016107.
- [22] J.J. Whang, P. Rai, I.S. Dhillon, Stochastic blockmodel with cluster overlap, relevance selection, and similarity-based smoothing, in: *Proceedings of IEEE International Conference on Data Mining*, 2013.
- [23] L. Lin, A concordance correlation coefficient to evaluate reproducibility, *Biometrika* 45 (1989) 255.
- [24] L. Lu, C.H. Jin, T. Zhou, Similarity index based on local paths for link prediction of complex networks, *Phys. Rev. E* 80 (2009) 046122.
- [25] L. Lu, T. Zhou, Y.C. Zhang, Predicting missing links via local information, *Eur. Phys. J. B* 71 (2009) 4.
- [26] H. Liao, A. Zeng, R. Xiao, D.B. Chen, Z.M. Ren, Y.C. Zhang, Ranking reputation and quality in online rating systems, *PLoS ONE* 9 (2014) e97146.
- [27] D. Lusseau, M.E.J. Newman, Identifying the role that individual animals play in their social network, *Behav. Ecol. Sociobiol.* 54 (2003) 396.
- [28] P.M. Gleiser, L. Danon, Community structure in JAZZ, *Adv. Complex Syst.* 6 (2003) 565.
- [29] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks, *Nature* 393 (1998) 440.
- [30] <http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm>.
- [31] M.E.J. Newman, Finding community structure in networks using the eigenvectors of matrices, *Phys. Rev. E* 74 (2006) 036104.
- [32] R. Guimera, L. Danon, A. Diaz-Guilera, F. Giralt, A. Arenas, Self-similar community structure in a network of human interactions, *Phys. Rev. E* 68 (2003) 065103.
- [33] A.C.C. Gavin, et al., Functional organization of the yeast proteome by systematic analysis of protein complexes, *Nature* 415 (2002) 141.
- [34] J.A. Hanley, B.J. Mcneil, The meaning and use of the area under a receiver operating characteristic (ROC) curve, *Radiology* 143 (1982) 29.
- [35] Y.X. Zhu, L.Y. Lu, Q.M. Zhang, T. Zhou, Uncovering missing links with cold ends, *Physica A* 391 (2012) 5769.
- [36] C. Spearman, The proof and measurement of association between two things, *Amer. J. Psychol.* 15 (1904) 72.
- [37] M. Kendall, A new measure of rank correlation, *Biometrika* 30 (1938) 81.
- [38] W. Zeng, A. Zeng, M.S. Shang, Y.C. Zhang, Information filtering in sparse online systems: Recommendation via semi-local diffusion, *PLoS ONE* 8 (2013) e79354.
- [39] T. Zhou, R.Q. Su, R.R. Liu, L.L. Jiang, B.H. Wang, et al., Accurate and diverse recommendations via eliminating redundant correlations, *New J. Phys.* 11 (2009) 123008.