epl A Letters Journal Exploring the Frontiers of Physics

January 2013

EPL, **101** (2013) 18004 www.epljournal.org
doi: 10.1209/0295-5075/101/18004

Hidden link prediction based on node centrality and weak ties

Haifeng Liu[1], Zheng Hu[1], Hamed Haddadi[2] and Hui Tian[1]

[1] *Key Laboratory of Universal Wireless Communications (Beijing University of Posts and Telecommunications), Ministry of Education - Beijing, China*
[2] *School of Electronic Engineering and Computer Science Queen Mary, University of London London, UK, EU*

received 30 September 2012; accepted in final form 20 December 2012
published online 18 January 2013

PACS 89.75.Hc – Networks and genealogical trees
PACS 89.20.Ff – Computer science and technology
PACS 89.65.-s – Social and economic systems

Abstract – Link prediction has been widely used to extract missing information, identify spurious interactions, evaluate network evolving mechanisms, and so on. In this context, similarity-based algorithms have become the mainstream. However, most of them take into account the contributions of each common neighbor equally to the connection likelihood of two nodes. This paper proposes a model for link prediction, which is based on the node centrality of common neighbors. Three node centralities are discussed: degree, closeness and betweenness centrality. In our model, each common neighbor plays a different role to the node connection likelihood according to their centralities. Moreover, the weak-tie theory is considered for improving the prediction accuracy. Finally, extensive experiments on five real-world networks show that the proposed model can outperform the Common Neighbor (CN) algorithm and gives competitively good prediction of or even better than Adamic-Adar (AA) index and Resource Allocation (RA) index.

Introduction. – Given a snapshot of a network at time t, which new links or interactions among its members are likely to occur at time $t'(t < t')$? We can formalize this question as the link prediction problem [1]. Link prediction is applicable to a variety of areas, such as protein-protein interaction (PPI) prediction [2], identifying spurious links [3], evaluation of network evolving mechanisms [4], e-commerce [5]. Zhou *et al.* [6] divided the link prediction algorithms into three categories: similarity-based algorithms, maximum-likelihood methods and probabilistic models. The similarity-based algorithms are the most used and they include node similarity and structural similarity.

This paper will focus on node similarity algorithms. Node similarity link prediction algorithms rely on the low complexity, low time consumption and good prediction accuracy, which become one of the most applied link prediction approaches. Among which, Common Neighbor (CN) [7] is the most widely used node-similarity-based algorithm. The basic assumption is that two nodes x and y are more likely to have a link if they have many common neighbors. CN only considers the number of common neighbors. Further, many variants [8–10] of CN are proposed by taking the degrees of nodes x and y into account. Therein, the Preferential Attachment (PA) index [4] is suitable for the prediction of scale-free networks, where the probability that a new link is connected to the nodes x and y is proportional to the degrees k_x and k_y. Furthermore Adamic-Adar [11] and Zhou *et al.* [12] improved the CN by restraining the contributions of large-degree common nodes. They further improved the prediction accuracy.

Most of the traditional approaches consider only the degree of each common neighbor of two nodes. They can improve the prediction accuracy, but, the improving is limited, because the node degree cannot reflect the significance of the node completely. Murata and Moriyasu [13] gave a weighted-common-neighbors approach. This paper assumed that proximities between nodes could be estimated better by using both graph proximity measures and the weights of existing links in a social network. It proposed a weighted graph proximity measures and new scores that took weights of links into account. Liu *et al.* [14] proposed a local naïve Bayes (LNB) model for link prediction in complex networks. In this model, different common neighbors will play different roles and give

different contributions. The proposed probabilistic model is based on the Bayesian theory. The connection probability of two nodes depends on the clustering coefficients of the common neighbors. And the authors of [14] proposed the improved LNB-CN, LNB-AA, LNB-RA according to the naïve Bayes model. In some networks, particularly social networks, weak ties play more important roles than strong ties. Lü and Zhou [15] provided the application of weak ties for link prediction in weighted networks.

In this paper, we propose a model which is based on the node centrality and weak-tie theory for link prediction. In our model, the significance of each common neighbor of two nodes is different according to their centralities. The model can be divided into two parts. In the first part, the centralities of all nodes will be computed. Three node centralities are considered: degree centrality, closeness centrality and betweenness centrality, to measure the connection probability between two nodes. Then, we combine the weak-tie theory with node centrality for improving the prediction accuracy. Finally, we conduct experiments on five real-world networks and compare with other four node-similarity-based link prediction algorithms: Common Neighbors (CN), Adamic-Adar (AA) index, Resource Allocation (RA) index and LNB [14]. The experimental results show that the proposed algorithm can outperform the CN and gives competitively good prediction of or even better than AA, RA and LNB.

The model based on node centrality and weak ties. – A network can be represented as $G(V, E)$, where V and E are the sets of nodes and links, respectively. Assuming $N = |V|$ denotes the number of nodes, $M = |E|$ denotes the number of links. $A = (a_{i,j})_{N \times N}$ is the adjacency matrix of the network. The multiple links and self-connections are not allowed. In this paper, we consider undirected and unweighted networks only.

Most of the similarity-based link prediction algorithms consider the contributions of each common neighbor of two nodes equally. In fact, each common neighbor may play different roles to the connection likelihood between two nodes in some social networks [14]. In this paper, we build a model based on the node centrality and weak-tie theory, which can treat each common neighbor as different contributions. Our model can be defined as follows:

$$s_{xy} = \sum_{z} \left(w\left(z\right) \cdot f\left(z\right)\right)^{\beta}, \qquad f\left(z\right) = \begin{cases} 1, & z \in \Gamma\left(x\right) \cap \Gamma\left(y\right), \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where $w\left(z\right)$ denotes the weight of node z_k, in this paper, and it represents the node centrality. $\Gamma\left(x\right)$ and $\Gamma\left(y\right)$ are the neighborhood of nodes x and y, respectively. The function of $f\left(z_k\right)$ is the switch function. Its value is unity if and only if the node z is the common neighbor of nodes x and y. The free parameter β can adjust the contributions of each common neighbor to the connection likelihood of the two nodes. If β is greater than one, it will amplify the contribution, otherwise, it can restrain the contribution.

Node centrality. The node significance is one of the most important research contents in social-network analysis. The significance represents the influence of a node. From the network perspective, the significance of an individual node is not an individual property, but arises from the node relations with other ones. Social-network analysis methods provide many useful tools for addressing one of the most important aspects of the social structure, such as the source and distribution of power [16]. The node centrality can measure the significance of the node. The position of the node in the network determines the ability to capture resource, information, to have more or less opportunities in favorable structure positions, and more or less imposing constraints. In this paper, we focus on three node centralities.

1) Degree Centrality.

Node degree refers to the number of connections or ties with other nodes. In undirected networks, it is equal to the neighbors of the nodes. The more ties a node has then, the more power or significance it may have. Nodes which have more ties have greater opportunities because they have more choices. This autonomy makes them less dependent on any specific other nodes, and hence more powerful. Because they have many ties, they may have access to, and be able to call on, more of the resources of the network as a whole [16]. So, a very simple, but very effective measure of a node centrality is its degree. The normalized degree centrality of node i is defined as follows:

$$DC_i = \frac{k_i}{N-1}, \tag{2}$$

where k_i is the degree of node i, N is the number of nodes. The higher DC_i is, the more central nodes i has.

2) Closeness Centrality.

Degree centrality measures might be criticized because they only take into account the immediate-vicinity ties that a node has, rather than indirect ties to all the others. Considering this case, one node may connect to a large number of other nodes, but these may be rather disconnected from the network as a whole. So, it is only in a central position in its local neighborhood. However, closeness centrality emphasizes the distance of one node to all others in the network by focusing on the geodesic distance from each node to all others. The sum of these geodesic distances for each node is the distance of the node from all the others. And this can be converted into a measure of closeness centrality [16]. Assuming d_i denotes the average distance from node i to all the others, d_i can be calculated as follows:

$$d_i = \frac{1}{N} \sum_{j=1}^{N} d_{ij}, \tag{3}$$

where d_{ij} denotes the distance between nodes i and j. The closeness centrality of node i can be defined as the reciprocal of d_i:

$$CC_i = \frac{1}{d_i} = \frac{N}{\sum_{j=1}^{N} d_{ij}}. \tag{4}$$

We can see that the node i is more significant if CC_i is larger. That is, if one node can access most of the other nodes as a small step, this node will have high influence.

3) Betweenness Centrality.

Betweenness centrality views a node as being in a favored position to the extent that the node falls on the geodesic paths between other pairs of nodes in the network [16]. That is to say, the more the nodes depend on a node to make connections with other nodes, the more power or significance that node has. Betweenness centrality represents the ability of capturing the flow of information. Freeman [17] gives the definition of betweenness centrality of node i as follows:

$$BC_i = \sum_{s \neq i \neq t} \frac{n_{st}^i}{g_{st}}, \tag{5}$$

where g_{st} denotes the number of shortest paths from node s to t, n_{st}^i is the number of g_{st} which pass node i. The normalized betweenness centrality is defined as follows:

$$BC_i = \frac{2}{N^2 - 3N + 2} \sum_{s \neq i \neq t} \frac{n_{st}^i}{g_{st}}. \tag{6}$$

We can also see that the higher BC_i is, the greater significance node i has.

Node-centrality-based algorithm. In this section, we introduce node-centrality-based link prediction algorithm. Assuming $\Gamma(x)$ and $\Gamma(y)$ denote the neighborhood of nodes x and y, respectively, x_i and y_j are the element of sets $\Gamma(x)$ and $\Gamma(y)$, respectively.

According to the CN algorithm, we can define three node-centralities-based CN algorithms: degree-centrality-based CN (DC-CN), closeness-centrality-based CN (CC-CN) and betweenness-centrality-based CN (BC-CN), which are called NC-CN algorithms. The formulas are defined as follows:

$$s_{xy}^{DC\text{-}CN} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} w_{DC}(z), \tag{7}$$

$$s_{xy}^{CC\text{-}CN} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} w_{CC}(z), \tag{8}$$

$$s_{xy}^{BC\text{-}CN} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} w_{BC}(z), \tag{9}$$

where z denotes the common neighbor of nodes x and y, $w_{DC}(z)$, $w_{CC}(z)$ and $w_{BC}(z)$ are the degree centrality, closeness centrality and betweenness centrality of node z, respectively.

From the three formulas we can see that the connection probability is the sum of the three node centralities, respectively. Because the node centralities of all nodes are different, the different common neighbors will have different contributions.

Combing weak ties with node centrality. For some networks, the weak ties play a more important role in the link prediction [15]. Onnela *et al.* [18] had shown that weak ties mainly maintain the network connectivity. In our experiments, we also find that the pure node-centrality-based link prediction algorithms (NC-CN) are not the best. In this section, we introduce a free parameter, β, to control the relative contributions of weak ties to the similarity measure. When β is greater than one, it makes the larger centrality more significant than the lower centrality. When β is less than zero, it restrains the larger centrality more than the lower centrality. When β is in the range $(0, 1)$, it equally restrains all nodes. It will become the CN, if β is equal to zero. The parameter-dependent indices for a node centrality based on the common-neighbor algorithm, which combines with the weak ties and is called NC-CN*, can be represented as DC-CN*, CC-CN* and BC-CN*, respectively:

$$s_{xy}^{DC\text{-}CN^*} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} (w_{DC}(z))^{\beta}, \tag{10}$$

$$s_{xy}^{CC\text{-}CN^*} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} (w_{CC}(z))^{\beta}, \tag{11}$$

$$s_{xy}^{BC\text{-}CN^*} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} (w_{BC}(z))^{\beta}. \tag{12}$$

These formulas show that the free parameter β can effectively control the contributions of the common neighbors of two nodes.

Experiments and results analysis. –

Data. We consider five real-world networks. The empirical data used in this paper include i) USAir [19]: The network of the US air transportation system, which contains 332 nodes and 2126 links; ii) NetScience(NS) [20]: A network of co-authorships between scientists who are themselves publishing on the topic of network science, the network has 1589 scientists and 2742 connections; iii) Power Grid [21]: An electrical power grid of the western US, with nodes representing generators, transformers and substations, and edges corresponding to the high voltage transmission lines between them, the network contains 4941 nodes and 6594 edges; iv) Yeast [2]: A protein-protein interaction network of yeast containing 2361 proteins and 6646 interactions; v) *C. elegans* (CE) [21]: The neural network of the nematode worm *C. elegans*, in which an edge joins two neurons if they are connected by either a synapse or a gap junction; the initial network includes many loops and multi-lines, for convenience, we eliminate all the loops and multi-lines; it contains 453 nodes and 2298 edges. Table 1 summarizes the basic

Table 1: The basic topological features of the five experimental networks. N and M denotes the total numbers of nodes and links, respectively. $\langle k\rangle$ is the average degree of the network. $\langle d\rangle$ is the average shortest distance between node pairs. C represents the clustering coefficient. DC, CC and BC is the degree centralization, closeness centralization and betweenness centralization of the whole network, respectively.

Networks	USAir	NS	Power	Yeast	CE
N	332	1589	4941	2361	453
M	2126	2742	6594	6646	2298
$\langle k\rangle$	12.807	3.451	2.669	5.63	8.94
$\langle d\rangle$	2.74	5.82	18.99	4.38	2.66
C	0.749	0.798	0.107	0.388	0.308
DC	0.383	0.019	0.003	0.025	0.507
CC	0.465	0.012	0.056	0.207	0.543
BC	0.204	0.022	0.285	0.037	0.476

topological features of the networks and the three node centralities. Wherein, DC, CC and BC are the degree closeness and betweenness centralization of the whole network, respectively. Special explanations can be seen in paper [16].

To test the algorithm's accuracy, all the network links, E, are randomly divided into two parts: the training set E^T, and the testing set, E^P. Obviously, $E = E^T \cup E^P$ and $E^T \cap E^P = \phi$. In our experiments, we adopt k-fold cross-validation. That is to say, we randomly divide all links into k subsets (in this paper, k is 10 according to [6]). Each time one subset is selected as testing set, the rest $k-1$ constitute the training set. The cross-validation process is then repeated k times, with each of the k subsets used exactly once as the testing set.

Two standard metrics are adopted in this paper, AUC [22] and precision [23], to quantify the accuracy of the prediction algorithms. AUC can be interpreted as the probability that a randomly chosen missing link (a link in E^P) is given a higher score than a randomly chosen non-existent link (a link in $U\backslash E$, where U denotes the universal link set). In the implementation, among n independent comparisons, if there are n' times, the missing link having a higher score, and n'' times in which the missing link and the non-existent link have the same score, AUC can be calculated:

$$AUC = \frac{n' + 0.5n''}{n}. \tag{13}$$

Precision is defined as the ratio of relevant links to the number of selected links. In our experiments, to calculate the precision, we firstly should rank all the missing and non-existent links in decreasing order according to their predicted scores. Then we focus on the top-L (here $L = 100$) links. If there are l links in the testing set E^P, then

$$Precision = {}^{l}/_{L}. \tag{14}$$

Results and analysis. In this section, we first show the performance of the three NC-CN algorithms with different β. Then, we compare our algorithms to other four similarity indices: Common Neighbor (CN), Adamic-Adar (AA) index, Resource Allocation (RA) index and the LNB [14].

i) CN [7]: For a node x, let $\Gamma(x)$ represents the neighborhood of x. Generally speaking, nodes x and y more probably have a link if they have more common neighbors. The simplest measure of this neighborhood overlap is the directed count, namely

$$s_{xy}^{CN} = |\Gamma(x) \cap \Gamma(y)|. \tag{15}$$

ii) AA [11]: This similarity measure refines the simple counting of common neighbors by giving the lower-degree neighbors more weights, as

$$s_{xy}^{AA} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k(z)}. \tag{16}$$

iii) RA [12]: It further restrains the contributions of large degree nodes, as

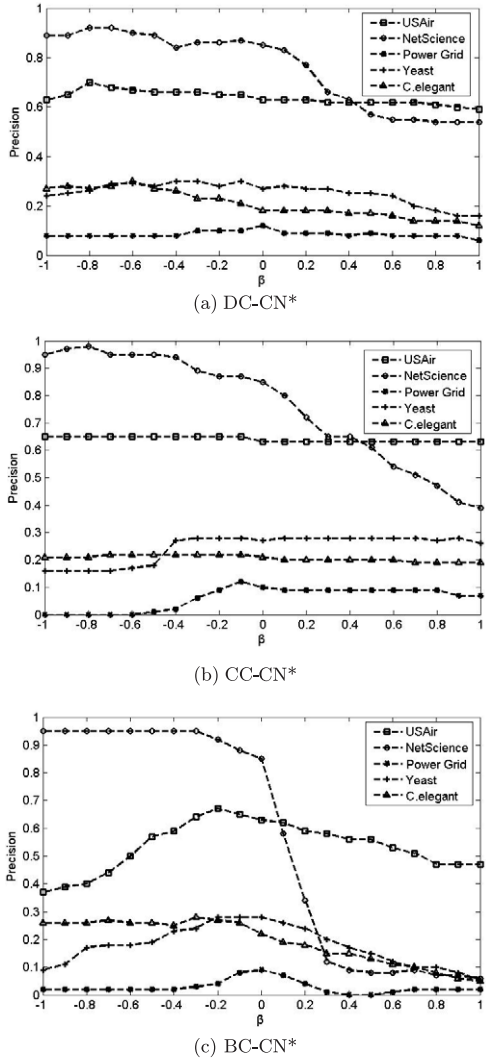
$$s_{xy}^{RA} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{k(z)}. \tag{17}$$

Just like many other papers [15,18], the top-L is set to 100 in our experiments. Figure 1 gives the variations of the node centrality based on the common-neighbor algorithm with different β. The x-axis denotes β and, the parameter $\beta \in [-1, 1]$. The y-axis represents the precision measure. In fig. 1(a), the best precision of DC-CN* is in the range $\beta < 0$ on all networks except the Power Grid. When β is equal to zero, the maximum precision can be captured on the Power Grid network. We also can find that the precision will decrease when β is greater than zero. Figure 1(b) shows the performance of the CC-CN* algorithm. We can see that the optimal value of β is negative on all five networks. However, the variance is not very obvious on Yeast and *C. elegans*. As we can see, fig. 1(c) gives the same variance with fig. 1(a). The difference is that the precision will decrease more sharp when the free parameter β is positive.

From fig. 1, we can see that the weak ties actually play a more important role than the strong links on some social networks, because the optimal precision values can be obtained when the parameter β is less than zero.

Figure 1 also indicates that the performance of the NC-CN* algorithm is related to the clustering coefficient (CC) of the network. From table 1 we can see that the CC of NetScience is the highest. Accordingly, the precision is also the best in fig. 1. The second is the USAir network. Its CC is close to NetScience one. But the little difference of CC will lead to a big difference in performance. Just like table 1, the CC of NetScience and USAir is 0.798 and 0.749, respectively. The difference is only 0.049. However, the precision of DC-CN* is 0.98 and 0.70, respectively. The precision decreases by 28% with CC reducing by 4.9% only.

Table 2 and table 3 give the comparisons at the AUC and precision measurements on five real-world networks,



(a) DC-CN*

(b) CC-CN*

(c) BC-CN*

Fig. 1: The performance of the three NC-CN* algorithms with different β on the five real networks. The x-axis and y-axis represents β and precision measure, respectively. The square curves, circle curves, star curves, plus curves and triangle curves denote the USAir, NetScience, Power Grid Yeast and *C. elegans* network, respectively. $L = 100$.

respectively. The bold face indicates the best value. Note that the results are at the optimal β. From table 2 we can see that the three pure node centrality link prediction algorithms (NC-CN) are worse than the traditional CN, AA, and RA. However, the three NC-CN* algorithms, which take the role of weak ties into account, outperform the CN except for the CC-CN* on Power Grid. The DC-CN* are even better than AA except for Yeast and better than RA in NetScience. Both CC-CN* and BC-CN* outperform AA and RA on Yeast and are equal to AA and RA on the Power Grid network, respectively. From the AUC perspective, the three NC-CN* algorithms are better than CN and partially outperform the AA and RA. And the difference is less than 0.01.

Table 3 also shows that the three pure node-centrality-based link prediction algorithms (NC-CN) are worse

Table 2: The prediction accuracy measured by AUC on five networks. n is ten millions. The abbreviations DC-CN*, CC-CN* and BC-CN* represent the highest precisions obtained by eq. (13), respectively.

Networks	USAir	NS	Power	Yeast	CE
CN	0.947	0.933	0.587	0.704	0.905
AA	0.959	0.934	0.587	0.705	0.950
RA	**0.964**	0.934	0.587	0.704	**0.959**
DC-CN	0.921	0.933	0.587	0.703	0.787
DC-CN*	0.963	**0.934**	0.587	0.704	0.955
CC-CN	0.942	0.933	0.509	0.704	0.872
CC-CN*	0.953	0.933	0.587	**0.705**	0.934
BC-CN	0.909	0.932	0.509	0.702	0.765
BC-CN*	0.957	0.934	**0.587**	0.704	0.951

Table 3: The prediction accuracy measured by the precision metric (top-100) on five networks. The abbreviations DC-CN*, CC-CN* and BC-CN* represent the highest precisions obtained by eq. (14), respectively.

Networks	USAir	NS	Power	Yeast	CE
CN	0.63	0.81	0.10	0.27	0.21
AA	0.66	0.94	0.07	**0.31**	0.29
RA	0.63	0.95	0.08	0.24	0.27
DC-CN	0.59	0.39	0.06	0.16	0.12
DC-CN*	**0.70**	**0.98**	0.10	0.30	**0.30**
CC-CN	0.63	0.54	0.07	0.26	0.19
CC-CN*	0.65	0.92	**0.12**	0.28	0.22
BC-CN	0.47	0.06	0.02	0.05	0.05
BC-CN*	0.67	0.95	0.10	0.28	0.28

than the CN, AA and RA. However, the three NC-CN* algorithms outperform the CN completely. DC-CN* and BC-CN* are also completely better than RA. DC-CN* is better than AA completely except on Yeast. BC-CN* is better than AA except for Yeast and *C. elegans*. CC-CN* is the best on Power Grid. It also outperforms RA except for NetScience and *C. elegans*. In general, the DC-CN* is the best in the three NC-CN* algorithms. Both table 2 and table 3 demonstrate that the NC-CN* algorithm can further improve the prediction of links. And, the performance is better when the cluster coefficient is high. Just like the NetScience and USAir networks, the precision further improves by 4% and 3% compared with AA and RA, respectively.

Table 4 and table 5 give the comparisons with the algorithms of paper [14]. From table 4, we can see that our algorithms outperform those of paper [14] on *C. elegans* and give competitive results on USAir. But, the AUC of NC-CN* is worse than the local naïve Bayes model. From table 5, we can also see that our algorithms are better than those of paper [14] on USAir and *C. elegans*, and are worse than those of paper [14] on Yeast. This may be because we eliminated loops and multi-edges in Yeast. From these two tables, we conclude that the proposed algorithm outperforms the algorithms of paper [14] on some networks. Especially, it improves more when the cluster coefficient of the network is very low. In the

Table 4: The comparison with paper [14] measured by AUC on three networks. The abbreviations DC-CN*, CC-CN* and BC-CN* represent the highest precisions obtained by eq. (13), respectively. LNB-CN, LNB-AA and LNB-RA are the local naïve-model-based algorithm [14], respectively.

Networks	USAir	Yeast	CE
LNB-CN	0.959	0.916	0.862
LNB-AA	0.967	0.916	0.866
LNB-RA	0.972	0.917	0.867
DC-CN*	0.9638	0.7048	0.9558
CC-CN*	0.9534	0.7051	0.9349
BC-CN*	0.9578	0.7049	0.9517

Table 5: The comparison with paper [14] measured by the precision metric (top-100) on three networks. The abbreviations DC-CN*, CC-CN* and BC-CN* represent the highest precisions obtained by eq. (14), respectively. LNB-CN, LNB-AA and LNB-RA are the local naïve-model-based algorithm [14] respectively.

Networks	USAir	Yeast	CE
LNB-CN	0.612	0.689	0.138
LNB-AA	0.629	0.703	0.136
LNB-RA	0.633	0.625	0.129
DC-CN*	0.70	0.30	0.30
CC-CN*	0.65	0.28	0.22
BC-CN*	0.67	0.28	0.28

same way, the precision can improve by 16.2% comparing DC-CN* with LNB-CN on *C. elegans.*

There are several conclusions to be drawn from these tables. First, we can see that the pure node centrality link prediction algorithms (NC-CN) are even worse than the traditional CN. Second, the algorithms that take into account the weak ties (NC-CN*) outperform the CN completely and AA and RA on some networks and give competitive results on other networks. Moreover, our algorithms are even better than those of paper [14] on some networks, especially the low-cluster-coefficient networks. Third, the DC-CN* achieves the best performance in the three node centralities link prediction algorithms (NC-CN*), followed by BC-CN* and CC-CN*. These results confirm that the node centrality combined with weak ties can further improve the prediction accuracy.

Conclusion. – This paper proposes a model based on node centrality and weak-ties theory for link prediction. To test our model, many experiments are implemented on five real-world networks and the results are compared to CN, AA and RA. The experiments show that the pure node-centrality-based algorithm is even worse than CN, AA and RA. However, it can outperform the CN completely and AA and RA on some networks, when the weak ties are combined with the node centrality link prediction algorithm. Moveover, we compare our algorithm with [14]. The results indicate that our algorithm can outperform LNB[14] in both AUC and precision, especially on the low-cluster-coefficient networks. These results demonstrate the effectiveness of the proposed model in link prediction.

* * *

This work has been funded by the national major projects (No. 2011ZX03005-004-02), and partially supported by Funds for Creative Research Groups of NSFC (No. 61121001) and Program for Changjiang Scholars and Innovative Research Team in MoE (No. IRT1049).

REFERENCES

[1] Liben-Nowell D. and Kleinberg J., *J. Am. Soc. Inf. Sci. Technol.*, **58** (2007) 1019.
[2] Bu D., Zhao Y., Cai L., Xue H., Zhu X., Lu H., Zhang J., Sun S., Ling L., Zhang N., Li G. and Chen R., *Nucleic Acids Res.*, **31** (2003) 2443.
[3] Guimerà R. and Sales-Pardo M., *Proc. Natl. Acad. Sci. U.S.A.*, **106** (2009) 22073.
[4] Barabási A. L. and Albert R., *Science*, **286** (1999) 509.
[5] Huang Z., Li X. and Chen H., *Link prediction approach to collaborative filtering*, in *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries* (ACM) 2005, pp. 141–142 .
[6] Lü L. and Zhou T., *Physica A: Stat. Mech. Appl.*, **390** (2011) 1150.
[7] Newman M. E. J., *Phys. Rev. E*, **64** (2001) 025102.
[8] Chowdhury G., *Introduction to Modern Information Retrieval* (Facet publishing) 2010.
[9] Leicht E. A., Holme P. and Newman M. E. J., *Phys. Rev. E*, **73** (2006) 026120.
[10] Ravasz E., Somera A. L., Mongru D. A., Oltvai Z. N. and Barabási A. L., *Science*, **297** (2002) 1551.
[11] Adamic L. A. and Adar E., *Soc. Netw.*, **25** (2003) 211.
[12] Zhou T., Lü L. and Zhang Y. C., *Eur. Phys. J. B*, **71** (2009) 623.
[13] Murata T. and Moriyasu S., *Link prediction of social networks based on weighted proximity measures*, in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence* (IEEE) 2007, pp. 85–88.
[14] Liu Z., Zhang Q. M., Lü L. and Zhou T., *EPL*, **96** (2011) 48007.
[15] Lü L. and Zhou T., *EPL*, **89** (2010) 18001.
[16] Hanneman R. A. and Riddle M., *Introduction to Social Network Methods* (University of California) 2005.
[17] Freeman L. C., *Sociometry*, **40** (1977) 35.
[18] Onnela J. P., Saramäki J., Hyvönen J., Szabó G., De Menezes M. A., Kaski K., Barabási A. L. and Kertész J., *New J. Phys.*, **9** (2007) 179.
[19] Batagelj V. and Mrvar A., `http://vlado.fmf.uni-lj.si/pub/networks/data/` (2006).
[20] Newman M. E. J., *SIAM Rev.*, **45** (2003) 175.
[21] Watts D. J. and Strogatz S. H., *Nature*, **393** (1998) 440.
[22] Hanley J. A. and McNeil B. J., *Radiology*, **148** (1983) 839.
[23] Herlocker J. L., Konstan J. A., Terveen L. G. and Riedl J. T., *ACM Trans. Inf. Syst.*, **22** (2004) 5.