Improving personalized link prediction by hybrid diffusion

Jin-Hu Liu\textsuperscript{a}, Yu-Xiao Zhu\textsuperscript{a,*}, Tao Zhou\textsuperscript{a,b}

\textsuperscript{a}Complex Lab, Web Sciences Center, University of Electronic Science and Technology of China, Chengdu 611731, China
\textsuperscript{b}Big Data Research Center, University of Electronic Science and Technology of China, Chengdu 611731, China

Abstract

Inspired by traditional link prediction and to solve the problem of recommending friends in social networks, we introduce the personalized link prediction in this paper, in which each individual will get equal number of diversiform predictions. While the performances of many classical algorithms are not satisfactory under this framework, thus new algorithms are in urgent need. Motivated by previous researches in other fields, we generalize heat conduction process to the framework of personalized link prediction and find that this method outperforms many classical similarity-based algorithms, especially in the performance of diversity. In addition, we demonstrate that adding one ground node who is supposed to connect all the nodes in the system will greatly benefit the performance of heat conduction. Finally, better hybrid algorithms composed of local random walk and heat conduction have been proposed. Numerical results show that the hybrid algorithms can outperform other algorithms simultaneously in all four adopted metrics: AUC, precision, recall and hamming distance. In a word, this work may shed some light on the in-depth understanding of the effect of physical processes in personalized link prediction.

Key words: Personalized link prediction, heat conduction, ground node

PACS: 89.65.-s, 89.75.Hc, 89.20.Ff

1. Introduction

Network has been used as a useful model to describe many social, biological and information systems, where nodes represent individuals and links reflect the relations or interactions between nodes \cite{1,2,3}. Networks have been widely studied in many different fields and one of fundamental problems for network analysis is link prediction, which aims to estimate the likelihood of the existence of a link between two nodes based on observed links and the attributes of nodes \cite{4,5}. For example, the existence of a link must be verified by costly chemical experiments in many biological networks, such as protein-protein interaction networks and metabolic networks. If the predictions are accurate enough, the experimental cost can be sharply reduced compared to blindly checking. Missing data problem is also exist in social network, where link prediction is also one useful tool. In addition, link prediction algorithms can also be applied to identify spurious links \cite{6,7,8}. Link prediction algorithms can not only be used to predict missing data but also practical to predict the links that may appear in the future of evolving networks. For

\textsuperscript{*}zhuyuxiao.mail@gmail.com(Yu-Xiao Zhu)

Preprint submitted to Physics A December 18, 2015
example, in online social networks, very likely but not yet existent links can be recommended as promising friendships, which can help users to find new friends and thus enhance their loyalties to the websites.

In the traditional link prediction, all the nonexistent links are sorted in descending order according to their prediction scores, and the top-ranked links are most likely to exist. Clearly, in this case, the prediction list is generated from a global perspective, in which some nodes may have large number of promising links while others may have very few or even zero possible links. This straightforward and standard method may lead to some bias. On the one hand, in this case, the links connect low-degree nodes may be ignored casually, while this kind of information may very be important and meaningful [9]. In addition, some research unveiled that low-degree users may have a big influence in the future [10]. On the other hand, the imbalance of prediction list may bring unsatisfactory for some individuals and thus affect the experience of the whole system. For example, in social networks, accurately predicting certain number of potential friends or acquaintances for each registered user is useful and meaningful. In this case, no real distinction can be made between low-degree and high-degree users and global link prediction does not apply in this case decently. However, this phenomenon has always been neglected in the traditional link prediction for the past several decades. To solve these problems, we propose personalized link prediction here, in which all nodes will get equal number of possible links through their own past link records.

One challenge deserved special attention recently, called low-diversity problem, has plagued almost all recommendation systems. It means that lots of recommender systems always recommend very similar items to different users which narrows users’ views [11]. Subsequently, some physical dynamics, like heat conduction process (HC) have been applied to design recommender systems and can improve the diversity of recommendation. Motivated by this, we generalize heat conduction process to the framework of personalized link prediction and find that it outperforms other methods in diversity but do not perform very satisfactorily in accuracy. To solve this dilemma, ground node, who is supposed to connect all the nodes in the system, is incorporated to improve the prediction accuracy. Finally, we generalize one superior hybrid algorithm (LH) and propose another better hybrid algorithm (LGH) composed of local random walk (LRW) [12] and ground heat conduction (GHC), which performs pretty well not only on accuracy but also on diversity.

This article is organized as follows. In the next section, we will clearly define the problem of personalized link prediction, describe the standard metrics for evaluation. Then we explain several state-of-the-art similarity indices and introduce new algorithms HC, GHC, LH, LGH in Section 3. Data description and experimental results for the existed predicting algorithms and the proposed method are presented in Section 4. Finally, we summarize our results in Section 5.

2. Problem and Metrics

For one given undirected network $G(V, E)$, in which $V$ and $E$ are the sets of nodes and links respectively. The universal set of all $\frac{|V|(|V|-1)}{2}$ possible links are denoted by $U$, where $|V|$ denotes the number of elements in set $V$ (multiple links and self-connections are not allowed). Clearly the set of nonexistent links is $U \setminus E$, in which there are some missing links (i.e., the existed yet unknown links) and promising links (i.e., very likely but not yet existent links). The task of link prediction is to uncover these links. Each node pair $x$ and $y$ will be assigned a score $s_{xy}$ according to a given prediction algorithm. The higher the score is, the higher existence likelihood this link has. For each node $x$, we denote the set of its relevant nonexistent links (nonexistent links that
connect x) as \((U \setminus E)_x\), thus all links in \((U \setminus E)_x\) are sorted in descending order according to their scores, and the top-ranked links are most likely to exist.

To test the performance of one given algorithm, we divide the observed links \(E\) into two sets: the training set \(E^T\) (considered as known information) and the test set \(E^P\) (used for testing and no information therein is allowed to be used for prediction). Clearly, \(E = E^T \cup E^P\) and \(E^T \cap E^P = \emptyset\). For each node \(x\), the relevant test set (links in \(E^P\) that connect \(x\)) is denoted by \(E^P_x\). We then introduce four popular evaluation metrics as below.

(i) **AUC** - short for area under the receiver operating characteristic curve, is considered as one standard metric to quantify the accuracy of prediction \([13]\). Specifically, for each node \(x\), this metric can be interpreted as the probability that a randomly chosen relevant missing link (links in \(E^P_x\)) has higher score than a randomly chosen relevant nonexistent link (links in \((U \setminus E)_x\)). In the implementation, among \(n\) times of independent comparisons, if there are \(n_1\) times that the missing link has higher score and \(n_2\) times the missing link and nonexistent link have the same score, the AUC value is defined as

\[
AUC = \frac{n_1 + 0.5n_2}{n}. \tag{1}
\]

The AUC of the whole system is the average value over all nodes in the system. If all the scores are generated from an independent and identical distribution, the accuracy should be about 0.5. Therefore, the extent to which the accuracy exceeds 0.5 indicates how much better the algorithm performs than pure chance.

(ii) **Precision and Recall** \([4]\) - Given the ranking of the non-observed links, the precision is defined as the ratio of relevant items selected to the number of items selected. Denoting by \(L\) the length of prediction list (i.e. the number of nodes recommended to each individual). For each individual \(x\), if we take the top-\(L\) links as the predicted ones, among which \(L_x\) links are right (i.e., there are \(L_x\) links in the test set \(E^P_x\)), then the precision equals \(P_x = L_x/L\). While recall is defined as the ratio of relevant items selected to the number of relevant items in the testing set. That’s, \(R_x = L_x/N_x\), where \(N_x\) denotes the number of node \(x\)’s positive edges in its testing set \(E^P_x\). Clearly, higher precision and higher recall means higher prediction accuracy. The precision (recall) of the whole system can be calculated by the average value among all individuals.

(iii) **Hamming distance** \([14, 15]\) - One of the famous metrics that quantify the intra-diversity of the prediction system. For individual \(x\) and \(y\), if the overlapped number of nodes in \(x\) and \(y\)’s prediction lists is \(Q_{xy}\), their Hamming distance is defined as

\[
H_{xy} = 1 - \frac{Q_{xy}}{L}. \tag{2}
\]

Generally speaking, a more diverse prediction list should have larger Hamming distances which means recommending appropriately but not popularly. Accordingly, we use the mean value of Hamming distance,

\[
H = \frac{1}{|\mathcal{V}|(|\mathcal{V}| - 1)} \sum_{x \neq y} H_{xy}, \tag{3}
\]

averaged over all the node-node pairs, to measure the diversity of predictions.
3. Algorithms

3.1. similarity-based algorithms

In the traditional link prediction problem, the study on similarity-based algorithms is the mainstream due to its simplicity. Considering this, we adopt the simplest local similarity indices as benchmark in the framework of personalized link prediction in this paper. For similarity-based algorithm, the aforementioned scores $s_{xy}$ is directly defined as the similarity between node $x$ and $y$ [16]. For each node $x$, rank all relevant links (links connect $x$ and other nodes) in relevant non-observed set $(U \setminus E)_x$ based on their scores, and links with higher scores are supposed to be of higher existence likelihoods and thus regarded as personalized prediction list (we consider length of list $L = 5, 20$). We will compare the performances of these indices on personalized link prediction and the details of these indices are as follows.

(1) Common Neighbors (CN). In common sense, two nodes $x$ and $y$ are more likely to have a link if they have many common neighbors [17]. The simplest measure of its neighborhood overlap is the direct count, namely

$$s_{xy}^{CN} = |\Gamma_x \cap \Gamma_y|,$$

in which $\Gamma_x$ denote the set of neighbors of $x$.

(2) Salton index [18]. It is defined as

$$s_{xy}^{Salton} = \frac{|\Gamma_x \cap \Gamma_y|}{\sqrt{k_x k_y}},$$

where $k_x$ denotes the degree of node $x$. The Salton index is also called the cosine similarity in the literature.

(3) Jaccard index [19]. This index was proposed by Jaccard over one hundred years ago, defined as

$$s_{xy}^{Jaccard} = \frac{|\Gamma_x \cap \Gamma_y|}{|\Gamma_x \cup \Gamma_y|}.$$

(4) Adamic Adar (AA) index [20]. This index refines the simple counting of common neighbors by assigning the less-connected neighbors more weights, and is defined as

$$s_{xy}^{AA} = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{\log k_z}.$$

(5) Preferential Attachment (PA) [21]. The mechanism of preferential attachment is widely used to generate evolving scale-free networks, where the probability that a new link is connected to the node $x$ is proportional to $k_x$. The corresponding similarity index can be defined as

$$s_{xy}^{PA} = k_x \times k_y.$$

(6) Resource Allocation (RA) index [22]. This index is motivated by the resource allocation dynamics on complex networks, and is defined as

$$s_{xy}^{RA} = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{k_z},$$

where $z$ runs over all common neighbors of $x$ and $y$. 
Local Random Walk (LRW) index [12]. To measure the similarity between nodes $x$ and $y$, a random walker is initially put on node $x$ and thus the initial density vector $\vec{\pi}_x(0) = \vec{e}_x$. This density vector evolves as $\vec{\pi}_x(t+1) = P^T \vec{\pi}_x(t)$ for $t \geq 0$. The LRW index at time step $t$ is thus defined as

$$s_{xy}^{LRW}(t) = q_x \pi_{xy}(t) + q_y \pi_{yx}(t),$$

where $q$ is the initial configuration function.

3.2. Heats Conduction algorithm and Hybrid algorithm

Motivated by some previous studies [13], here we generalize the heat conduction algorithm to the framework of personalized link prediction. Basically, heats conduction algorithm (HC) recommends promising links to an individual node through a process motivated by heat diffusion. Firstly, the adjacency matrix, denoted by $A$, where the element $a_{xy} = 1$ if node $x$ has connection with node $y$, $a_{xy} = 0$ otherwise. Let assume one target node $x$ here, the temperature resource for $x$ is initialized with 1, while 0 for the remaining nodes. Then in each iteration, the HC algorithm redistributes the temperature resource via a nearest-neighbor averaging process. That is to say, the temperature resource of node $x$ in $(t+1)$th iteration can be written as:

$$T_x(t+1) = \frac{1}{k_x} \sum_{y=1}^{M} a_{xy} T_y(t),$$

where $k_x$ is the degree of node $x$. The numerical value over each node indicates its temperature resource in this step. The red node is the target node. In each step after step 2, those high-temperature nodes which have no links with the target node will be recommended.

Figure 1: The schematic drawing that illustrates LRW and HC from Step:0 to Step:2. Nodes are represented by circles and lines represent the existent links between them. The numerical value over each node indicates its temperature resource in this step. The red node is the target node. In each step after step 2, those high-temperature nodes which have no links with the target node will be recommended.
where $M$ denotes the number of nodes. The degree of node $x$, the number of nodes who have connections with $x$, is denoted as $k_x$. Then the final temperatures after the diffusion are considered as the corresponding scores and the resulting top-ranked list of non-connected nodes is sorted according to these scores in descending order. In order to describe the detailed diffusion process visually, an illustration of the first two step of HC and LRW processes is shown in Figure [1].

In depth, we also propose a new algorithm by adding a ground node who connect all the nodes in the network [23]. The iterative temperature of the GHC (abbreviation of the HC with ground node) algorithm is thus written as

$$T_x(t+1) = \frac{1}{k_x + 1} \sum_{y=1}^{M} a_{xy} T_y(t) + T_g(t).$$

(12)

where $T_g$ denotes the temperature of the ground node, which leads to an additional link between two nodes even when they don’t have connection.

In addition, motivated by the literatures [11, 24], we generalize one superior hybrid algorithm (LH) composed of LRW and HC.

$$T_x(t+1) = \alpha \sum_{y=1}^{M} \frac{a_{xy} T_y(t)}{k_y} + \frac{1 - \alpha}{k_x + 1} \sum_{y=1}^{M} a_{xy} T_y(t).$$

(13)

where $\alpha$ is an adjustable parameter which ranges from 0 to 1. Obviously, LH turns to HC when $\alpha = 0$, while degenerates to LRW when $\alpha = 1$. This method can well test potential nodes from two aspects, one is the strength of joint and the other is the personalization. Furthermore, we propose another novel hybrid algorithm (LGH) by combining LRW and GHC.

$$T_x(t+1) = \alpha \sum_{y=1}^{M} \frac{a_{xy} T_y(t)}{k_y} + \frac{1 - \alpha}{k_x + 1} \sum_{y=1}^{M} a_{xy} T_y(t) + T_g(t).$$

(14)

where $\alpha$ is an adjustable parameter which ranges from 0 to 1. Obviously, LGH turns to GHC when $\alpha = 0$, while degenerates to LRW when $\alpha = 1$.

4. Data and Experiments

4.1. Data

We consider four representative networks drawn from disparate fields: (i) USAir. The network of the USAir transportation system, which contains 332 airports and 2126 airlines [25]. (ii) C.elegans (CE). 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 [26]. This network contains 297 neurons and 2148 links. (iii) Political Blogs (PB). The network of US political blogs [27]. The original links are directed, here we treat them as undirected links. (iv) Food Webs (FW). The food circle network of Florida bay, containing 128 living beings and 2106 preying relations. Table [1] summarizes the basic topological features of these networks [28]. Brief definitions of the monitored topological measures can be found in the table caption.
Table 1: The basic topological features of the giant components of the four example networks. CE, PB and FW are the abbreviations for C.elegans, Political Blogs and Food Webs networks, respectively. \( N = |V| \) and \( M = |E| \) are the total number of nodes and links, respectively. \( C \) is the clustering coefficient that is defined as the average ratios of the number of connected pairs of a node’s neighbors to the possible maximum \[29\]. \( r \) is the assortative coefficient \[30\], the Pearson correlation coefficient of degree between pairs of connected nodes. \( r \) lies between \(-1\) and \(1\), and \( r > 0 \) indicates a positive correlation while \( r < 0 \) indicates a negative correlation. \( \langle k \rangle \) is the average degree of the network, and \( \langle d \rangle \) is the average shortest distance between node pairs. \( H \) denotes the degree heterogeneity defined as \( H = \frac{\langle k^2 \rangle}{\langle k \rangle^2} \).

| Datasets | \( N \) | \( M \) | \( C \) | \( r \) | \( \langle k \rangle \) | \( \langle d \rangle \) | \( H \) |
|----------|-------|-------|-------|-------|-------------|-------------|-------|
| USAir    | 322   | 2126  | 0.749 | -0.208| 12.81       | 2.46        | 3.46  |
| CE       | 297   | 2148  | 0.308 | -0.163| 14.46       | 2.46        | 1.80  |
| PB       | 1222  | 16717 | 0.361 | -0.221| 27.36       | 2.51        | 2.97  |
| FW       | 128   | 2106  | 0.335 | -0.104| 32.90       | 1.77        | 1.23  |

4.2. Experimental Results

To test the performances of algorithms, we divide the data sets into training set and testing set randomly in our experiments. The ratio of training set to testing set is 9:1, that is to say, testing set contains 10% links. The prediction list for each individual is provided based on the training set, and the testing set will be used for testing. Four metrics like AUC, precision, recall and hamming distance are adopted here to give quantitative measurements of the methods. All the results below are averaged over 100 independent runs with different data divisions.

Table 2: Algorithmic accuracies on four different datasets, measured by AUC. Each value is obtained by averaging over 100 implementations with independently divisions of training set and test set randomly. CE, PB and FW are the abbreviations for C.elegans, Political Blogs and Food webs networks respectively. The number in bracket indicates the step in which the corresponding algorithm gets optimal value.

| Datasets | USAir | CE     | PB     | FW     |
|----------|-------|--------|--------|--------|
| CN       | 0.9354| 0.8285 | 0.8645 | 0.5516 |
| Salton   | 0.9049| 0.7979 | 0.8129 | 0.4988 |
| Jaccard  | 0.8853| 0.7933 | 0.8018 | 0.4988 |
| AA       | 0.9432| 0.8405 | 0.8688 | 0.5589 |
| PA       | 0.8945| 0.7724 | 0.8499 | 0.6771 |
| RA       | 0.8976| 0.8440 | 0.8697 | 0.5656 |
| LRW      | 0.9409(4) | 0.8821(3) | 0.9153(3) | 0.8412(3) |
| HC       | 0.8450(3) | 0.8548(3) | 0.8426(3) | 0.8804(3) |
| GHC      | 0.8697(3) | 0.8614(3) | 0.8651(3) | 0.8821(3) |
| LH       | 0.9507(4) | 0.8961(3) | 0.9189(3) | 0.9086(3) |
| LGH      | 0.9539(4) | 0.8973(3) | 0.9204(3) | 0.9091(3) |

The performances of different methods measured by AUC in all data sets are shown in Table 2. Highest AUC value in each column is emphasized in black. The optimal step of iterations
Figure 2: The performance of LH and LGH measured by four metrics (AUC, precision (P), recall (R) and hamming distance (H)) as a function of $\alpha$ in the representative data sets. The blue and black curves represent results of LH and LGH in the step getting optimal AUC, respectively. And the red solid line represents results of LRW in the step getting optimal AUC. All the numerical results are obtained by averaging over 100 independent runs with data division. We set $L = 5$ here.
Figure 3: The performance of LH and LGH measured by four metrics (auc, precision (P), recall (R) and hamming distance (H)) as a function of $\alpha$ in the representative data sets. The blue and black curves represent results of LH and LGH in the step getting optimal AUC, respectively. And the red solid line represents results of LRW in the step getting optimal AUC. All the numerical results are obtained by averaging over 100 independent runs with data division identical to the case shown in Figure 2. We set $L = 20$ here.
correspond to best AUC are shown in the brackets. GHC is an abbreviation of the method HC with a ground node. LH refers to the hybrid method that combines LRW and HC algorithms, while LGH refers to the hybrid method that combines LRW and GHC algorithms. Clearly, HC algorithm outperforms classical benchmark algorithms (CN, Salton, Jaccard, AA and PA) in most networks, while is slightly inferior to LRW. In order to improve the prediction accuracy of HC, we propose one new method called GHC by adding one ground node in the system. By comparing the results of HC and GHC, we can see that the prediction accuracy can be improved to some extent by this way. For example, the AUC increases from 0.8450 to 0.8697 for USAir data set. Previous studies have shown that the original HC algorithm prefers to the small-degree nodes [11]. Adding one ground node to the system practically amounts to add an additional transition probability from ground node to another. So in every iteration, each node receives the same heat from the ground node and then average it with the heat from other sources, thus the temperature of the big-degree nodes will be enhanced. This improvement of the accurate predictions on big-degree nodes leads to the improvement of prediction accuracy of the whole system. Moreover, diffusion-based algorithms will be restricted by the network connectedness. The ground node happens to make the whole network much more strongly connected and the shortest path between any two nodes is less than 3. That’s why GHC performs better than HC.

![Figure 4](image.png)

**Figure 4:** The correlation between AUC and the diffusion step. The black squares, red circles, blue triangles, cyan triangles and wine stars represent the AUC of the five algorithms in the each diffusion step, respectively. The values of AUC for LH and LGH are all corresponding to optimal values in each step.

Furthermore, in Figure, the black and blue curves show the results of LGH and LH with the corresponding $\alpha$ respectively, for comparison, results of LRW ($\alpha = 1$) are displayed in red solid line. Obviously, hybrid algorithms have greatly improved the prediction results. By combining LRW and GHC, the hybrid algorithm (LGH) performs the best in the accuracy of
prediction measured by AUC. Besides, Figure 2 displays the performances of LGH measured by recall and precision in relationship with free parameter $\alpha$, although its diversity is a little lower than LH. The advantage of hybrid algorithms comes from two aspects. Firstly, LRW makes those popular nodes still having determinate weighting to exist in the prediction list. Secondly, more importantly, GHC or HC enhances the weights of those low-degree nodes and reduces the weights of high-degree nodes simultaneously, which make the prediction lists much more personalized and accurate. The hybrid algorithms with certain range of free parameter not only improve the accuracy (AUC, precision, recall) but also benefit the diversity of prediction. The improvements of this hybrid algorithms are independent of prediction length $L$. Figure 3 shows the corresponding results when $L = 20$, which display the same tendency.

For all diffusion-based method, one key problem is the diffusion step. Liu et al. have already proven a positive correlation between the optimal step and the average shortest distance [12]. In our algorithm, the average shortest distance of a network is getting smaller with the adding of ground node. So our algorithm can quickly obtain the optimal value with less diffusion steps. From Table 2 and Figure 4, it is clearly that LH and LGH both obtain their best results within four steps in all the four data sets. And after the 5th step, the results in all data sets get worse with the increase of diffusion step.

5. Conclusion and Discussion

In the past several decades, Internet is flourishing as never before. Especially, as new free-registered platforms, social networks provide us with abundant information, countless items and infinite opportunities to make fresh friends around the world. Meanwhile, we are in deep trouble when facing information overload. We are extremely hard to dig out interesting information, suitable items to purchase, new friends with the same abilities and interests and old acquaintances we lost touch with. Fortunately, the appearance of personalized recommendation and prediction gives us considerable assistance. Based on historical behaviors, potential interests and links will be found out automatically via appropriate algorithms. Thus, an accurate and effective algorithm is vital and extremely valuable. But to our knowledge, in the previous studies on link prediction, all the nonexistent links are sorted in descending order according to their prediction scores, and the top-ranked links are most likely to exist. Clearly, in this case, the top-ranked list is generated from a global perspective, in which some nodes may have large number of promising links while others may have very few or even zero possible links. Considering this bias, we propose personalized link prediction problem in this paper, in which all nodes will get equal number of possible links. Motivated by some previous works, we generalized the physical process - heat conduction to the framework of personalized link prediction and find that it outperforms most existing similarity-based algorithms. In addition, we demonstrate that adding one ground node which is supposed to connect all the nodes in the system will benefit the performance of HC. Finally, we introduce two hybrid algorithms that perform pretty well not only on accuracy but also diversity. Note that, one small defect of our algorithms is relatively complicated and time-cost, fortunately this weakness can be easily solved by parallel computation technique. With the improvement of calculating technique, an algorithm with better performance must be a top choice.

However, how to provide better personalized predictions is still a long-standing challenge in modern information science. One satisfying answer to this question may benefit our society, economic and lifestyle in the near future. As a starting point, we give a naive method and a
preliminary analysis, which is of course far from a satisfactory answer to the question. In fact, we believe the current work can provide some insights to understand this issue.

6. Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under grant nos. 61370150 and 61433014, and the Program of Outstanding PhD Candidate in Academic Research by UESTC. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

[1] M. E. J. Newman, The structure and function of complex networks, SIAM Rev. 45 (2) (2003) 167–256.
[2] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.-U. Hwang, Complex networks: Structure and dynamics, Phys. Rep. 424 (4) (2006) 175–308.
[3] R. Cohen, S. Havlin, Complex Networks: Structure, Robustness and Function, Cambridge University Press, 2010.
[4] L. Y. Li, T. Zhou, Link prediction in complex networks: a survey, Physica A 390 (6) (2011) 1150–1170.
[5] L. Getoor, C. P. Diehl, Link mining: a survey, ACM SIGKDD Explorations Newsletter 7 (3) (2005) 3–12.
[6] G. Kossinets, Effects of missing data in social networks, Social Networks 28 (2006) 247–268.
[7] R. Guimerà, M. Sales-Pardo, Missing and spurious interactions and the reconstruction of complex networks, Proc. Natl. Acad. Sci. U.S.A. 106 (52) (2009) 22073–22078.
[8] A. Zeng, G. Cimini, Removing spurious interactions in complex networks, Phys. Rev. E 85 (3) (2012) 036101.
[9] Y. X. Zhu, L. Y. Liu, Q. M. Zhang, T. Zhou, Uncovering missing links with cold ends, Physica A 391 (22) (2012) 5769–5778.
[10] J.-H. Liu, T. Zhou, Z.-K. Zhang, Z. Yang, C. Liu, W.-M. Li, Promoting cold-start items in recommender systems, PLoS One 9 (12) (2014) e113457.
[11] T. Zhou, Z. Kucskik, J. G. Liu, M. Medo, J. R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proc. Natl. Acad. Sci. U.S.A. 107 (10) (2010) 4511–4515.
[12] W. Liu, L. Lü, Link prediction based on local random walk, EPL 89 (5) (2010) 58007.
[13] J. A. Hanley, B. J. McNeil, The meaning and use of the area under a receiver operating characteristic (roc) curve, Radiology 143 (1) (1982) 29–36.
[14] T. Zhou, L.-L. Jiang, R.-Q. Su, Y.-C. Zhang, Effect of initial configuration on network-based recommendation, EPL 81 (5) (2008) 58004.
[15] T. Zhou, R.-Q. Su, R.-R. Liu, L.-L. Jiang, B.-H. Wang, Y.-C. Zhang, Accurate and diverse recommendations via eliminating redundant correlations, New J. Phys. 11 (12) (2009) 123008–123026.
[16] D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks, J. Am. Soc. Inform. Sci. Technol. 58 (7) (2007) 1019–1031.
[17] G. Kossinets, Effects of missing data in social networks, Social Networks 28 (2006) 247–268.
[18] G. Salton, M. J. McGill, Introduction to Modern Information Retrieval, MuGraw-Hill, Auckland, 1983.
[19] P. Jaccard, ‘Etude comparative de la distribution florale dans une portion des alpes et des jura, Bulletin de la Societe Vaudoise des Science Naturelles 37 (1901) 547–579.
[20] L. A. Adamic, E. Adar, Friends and neighbors on the web, Social Networks 25 (2003) 211–230.
[21] A. L. Barabási, R. Albert, Emergence of scaling in random networks, Science 286 (1999) 509–512.
[22] T. Zhou, L. Lü, Y.-C. Zhang, Predicting missing links via local information, Eur. Phys. J. B 71 (2009) 623–630.
[23] L. Y. Li, Y.-C. Zhang, C. H. Yeung, T. Zhou, Leaders in social networks, the delicious case, PLoS One 6 (6) (2011) e21202.
[24] R.-R. Liu, J.-G. Liu, C.-X. Jia, B.-H. Wang, Personal recommendation via unequal resource allocation on bipartite networks, Physica A 389 (16) (2010) 3282–3289.
[25] V. Batagelj, A. Mrvar, Pajek datasets, http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm
[26] J. E. White, E. Southgate, I. N. Thomson, S. Brenner, Philos. Trans. R. Soc. B 314 (1).
[27] R. Ackland, Mapping the us political blogosphere: Are conservative bloggers more prominent?, URL http://incsub.org/blogtalk/images/robertackland.pdf
[28] R. E. Ulanowicz, C. Bondavalli, M. S. Egnovitch, Technical report, cbl (98-125) (1998).
URL http://www.cbl.umces.edu/atlas/EBay701.html.
[29] D. J. Watts, S. Strogatz, Collective dynamics of small-world networks, Nature 393 (1998) 440–442.
[30] M. E. J. Newman, Assortative mixing in networks, Phys. Rev. Lett. 89 (2007) 208701.