Physica A 390 (2011) 1150–1170

Contents lists available at ScienceDirect

Physica A

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

Minireview

Link prediction in complex networks: A survey

Linyuan Lü [a,b,c], Tao Zhou [a,d,*]

[a] Web Sciences Center, University of Electronic Science and Technology of China, Chengdu 610054, People's Republic of China
[b] Research Center for Complex System Science, University of Shanghai for Science and Technology, Shanghai 200093, People's Republic of China
[c] Department of Physics, University of Fribourg, Chemin du Musée 3, CH-1700 Fribourg, Switzerland
[d] Department of Modern Physics, University of Science and Technology of China, Hefei 230026, People's Republic of China

ARTICLE INFO

Article history:
Received 5 October 2010
Received in revised form 10 November 2010
Available online 2 December 2010

Keywords:
Link prediction
Complex networks
Node similarity
Maximum likelihood methods
Probabilistic models

ABSTRACT

Link prediction in complex networks has attracted increasing attention from both physical and computer science communities. The algorithms can be used to extract missing information, identify spurious interactions, evaluate network evolving mechanisms, and so on. This article summaries recent progress about link prediction algorithms, emphasizing on the contributions from physical perspectives and approaches, such as the random-walk-based methods and the maximum likelihood methods. We also introduce three typical applications: reconstruction of networks, evaluation of network evolving mechanism and classification of partially labeled networks. Finally, we introduce some applications and outline future challenges of link prediction algorithms.

Contents

* Corresponding author at: Web Sciences Center, University of Electronic Science and Technology of China, Chengdu 610054, People's Republic of China.
E-mail addresses: linyuan.lue@unifr.ch (L. Lü), zhutou@ustc.edu (T. Zhou).

0378-4371/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.physa.2010.11.027

1. Introduction

Many social, biological, and information systems can be well described by networks, where nodes represent individuals, biological elements (proteins, genes, etc.), computers, web users, and so on, and links denote the relations or interactions between nodes. The study of complex networks has therefore become a common focus of many branches of science. Great efforts have been made to understand the evolution of networks [1,2], the relations between topologies and functions [3,4], and the network characteristics [5]. An important scientific issue relevant to network analysis is the so-called *information retrieval* [6,7], which aims at finding material of an unstructured nature that satisfies an information needed from large collections [8]. It can also be viewed as prediction of relations between words and documents and is now further extended to stand for a number of problems on link mining, wherein *link prediction* is the most fundamental problem that attempts to estimate the likelihood of the existence of a link between two nodes, based on observed links and the attributes of nodes [9].

In many biological networks, such as food webs, protein–protein interaction networks and metabolic networks, whether a link between two nodes exists must be demonstrated by field and/or laboratorial experiments, which are usually very costly. Our knowledge of these networks is very limited, for example, 80% of the molecular interactions in cells of Yeast [10] and 99.7% of human [11,12] are still unknown. Instead of blindly checking all possible interactions, to predict based on known interactions and focus on those links most likely to exist can sharply reduce the experimental costs if the predictions are accurate enough. Social network analysis also comes up against the missing data problem [13–15], where link prediction algorithms may play a role. In addition, the data in constructing biological and social networks may contain inaccurate information, resulting in spurious links [16,17]. Link prediction algorithms can be applied in identifying these spurious links [18]. Readers should be warned that some "unexpected" links may be incorrectly identified as spurious links and thus the removal of these links may lead to biased understanding of the system's structure and function. Actually, as we will show in Section 6.1, the method by Guimerà and Sales-Pardo [18] can find out most of the spurious links yet incorrectly remove some real links. As a whole we believe that these kinds of methods are helpful because the reconstructed network is shown to have closer functionality to the real network.

Besides helping in analyzing networks with missing data, the link prediction algorithms can be used to predict the links that may appear in the future of evolving networks. For example, in online social networks, very likely but not-yet-existent links can be recommended as promising friendships, which can help users in finding new friends and thus enhance their loyalties to the web sites. Similar techniques can be applied to evaluate the evolving mechanism for given networks. For example, many evolving models for the Internet topology have been proposed: some more accurately reproduce the degree distribution and the disassortative mixing pattern [19], some better characterize the k-core structure [20], and so on. Since there are too many topological features and it is very hard to put weights on them, we are not easy to judge which model (i.e., which evolving mechanism) is better than the others. Note that, each model in principle corresponds to a link prediction algorithm, and thus we can use the metrics on prediction accuracy to evaluate the performance of different models.

Link prediction problem is a long-standing challenge in modern information science, and a lot of algorithms based on Markov chains and statistical models have been proposed by computer science community. However, their works have not caught up the current progress of the study of complex networks, especially, they lack serious consideration of the structural characteristics of networks, like the hierarchical organization [21] and community structure [22], which may indeed provide useful information and insights for link prediction. Recently, some physical approaches, such as random walk processes and maximum likelihood methods, have found applications in link prediction. This article will give detailed discussion on these new developments.

This article is organized as follows. In the next section, we will present the link prediction problem and the standard metrics for performance evaluation. Our tour of link prediction algorithms starts with the mainstreaming class of algorithms, the so-called *similarity-based algorithms*,[1] which are further classified into three categories according to the information used by the similarity indices: local indices, global indices and quasi-local indices. In Sections 4 and 5, we introduce the maximum likelihood algorithms and probabilistic models for link prediction. The applications of link prediction algorithms are presented in Section 6, including the reconstruction of networks, the evaluation of network evolving mechanism and the classification of partially labeled networks. Finally, we outline some future challenges of link prediction algorithms.

2. Problem description and evaluation metrics

Consider an undirected network $G(V, E)$, where V is the set of nodes and E is the set of links. Multiple links and self-connections are not allowed.[2] Denote by U, the universal set containing all $\frac{|V|\cdot(|V|-1)}{2}$ possible links, where $|V|$ denotes the number of elements in set V. Then, the set of nonexistent links is $U - E$. We assume that there are some missing links (or the links that will appear in the future) in the set $U - E$, and the task of link prediction is to find out these links.

Generally, we do not know which links are the missing or future links, otherwise we do not need to do prediction. Therefore, to test the algorithm's accuracy, the observed links, E, is randomly divided into two parts: the training set, E^T,

[1] The similarity indices between nodes are also called kernels on graphs in some literature of computer science community [23].

[2] A network with multiple links can be represented by a weighted network where the weight of a link connecting two nodes equals the number of links between these two nodes [24]. We will discuss the problem of link prediction on weighted networks in Section 7.

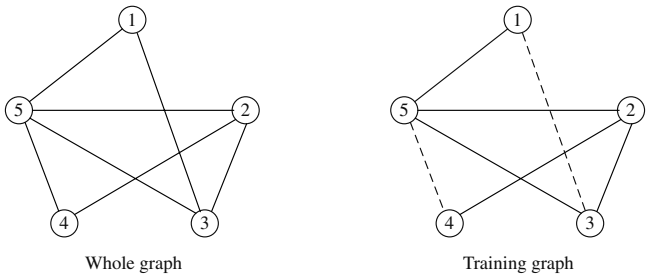

Fig. 1. An illustration about the calculation of AUC and precision.

is treated as known information, while the probe set (i.e., validation subset), E^P, is used for testing and no information in this set is allowed to be used for prediction. Clearly, $E^T \cup E^P = E$ and $E^T \cap E^P = \emptyset$. The advantage of this random sub-sampling validation is that the proportion of the training split is not dependent on the number of iterations. But with this method some links may not be selected in the probe set, whereas others may be selected many times. This will lead to some statistical errors. Such a disadvantage can be overcome by using the *K-fold cross-validation*, in which the observed links are randomly partitioned into K subsets. Each time one subset is selected as probe 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 probe set. With this method, all links are used for both training and validation, and each link is used for prediction exactly once. Clearly, a larger K will lead to smaller statistical bias yet require more computation. Some experimental evidences suggested that the 10-fold cross-validation is a very good tradeoff between cost and performance [25,26]. An extreme case called *leave-one-out* method (i.e., the $|V|$-fold cross-validation) will be applied in Section 6.2. Note that, the random fashion of the division of the training and probe sets may fail to capture the problem of some real systems. Actually, the missing links are more likely to be the ones connecting low-degree nodes [27]. For example, in a metabolic network, the chemical relations between main reactants, like H_2O, CO_2, NH_3, O_2, GTP, and so on, are well known to the scientific community, while the missing interactions are usually the ones between infrequently observed reactants. In addition, missing links of sampled networks, like the Internet and the World Wide Webs, are always the ones connecting marginal nodes.

Two standard metrics are used to quantify the accuracy of prediction algorithms: *area under the receiver operating characteristic curve* (AUC)[3] [30] and *Precision* [31,32]. In principle, a link prediction algorithm provides an ordered list of all non-observed links (i.e., $U - E^T$) or equivalently gives each non-observed link, say $(x, y) \in U - E^T$, a score s_{xy} to quantify its existence likelihood. The AUC evaluates the algorithm's performance according to the whole list while the precision only focuses on the L links with top ranks or highest scores. A detailed introduction of these two metrics is as follows.

(i) AUC: Provided the rank of all non-observed links, the AUC value can be interpreted as the probability that a randomly chosen missing link (i.e., a link in E^P) is given a higher score than a randomly chosen nonexistent link (i.e., a link in $U - E$). In the algorithmic implementation, we usually calculate the score of each non-observed link instead of giving the ordered list since the latter task is more time consuming.[4] Then, at each time we randomly pick a missing link and a nonexistent link to compare their scores, if among n independent comparisons, there are n' times the missing link having a higher score and n'' times they have the same score, the AUC value is

$$\mathrm{AUC} = \frac{n' + 0.5n''}{n}. \tag{1}$$

If all the scores are generated from an independent and identical distribution, the AUC value should be about 0.5. Therefore, the degree to which the value exceeds 0.5 indicates how better the algorithm performs than pure chance.

(ii) Precision: 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. That is to say, if we take the top-L links as the predicted ones, among which L_r links are right (i.e., there are L_r links in the probe set E^P), then the precision equals L_r/L. Clearly, higher precision means higher prediction accuracy.

Fig. 1 shows an example of how to calculate the AUC and precision. In this simple graph, there are five nodes, seven existent links and three nonexistent links ((1, 2), (1, 4) and (3, 4)). To test the algorithm's accuracy, we need to select some existent links as probe links. For instance, we pick (1, 3) and (4, 5) as probe links, which are presented by dash lines in the right plot. Then, an algorithm can only make use of the information contained in the training graph (presented by solid lines in the right plot). If an algorithm assigns scores of all non-observed links as $s_{12} = 0.4$, $s_{13} = 0.5$, $s_{14} = 0.6$, $s_{34} = 0.5$ and

[3] Actually, AUC is formally equivalent to the Mann–Whitney U statistical test (or Wilcoxon rank-sum test) which is a non-parametric test for assessing whether two independent samples of observations come from the same distribution [28,29]. For this metric, a basic assumption is that all the links are independent, which may not be correct in real networks.

[4] The computational complexity for an ordered list of nonexistent links in a sparse network is $\mathcal{O}(|V|^2 \log |V|^2)$. Since the number of nodes $|V|$ can be very large, it is very time consuming. As shown in Eq. (1), instead of the exact value, to estimate the AUC value with very good accuracy does not need to know the ordered list.

$s_{45} = 0.6$. To calculate AUC, we need to compare the scores of a probe link and a nonexistent link. There are six pairs in total: $s_{13} > s_{12}, s_{13} < s_{14}, s_{13} = s_{34}, s_{45} > s_{12}, s_{45} = s_{14}$ and $s_{45} > s_{34}$. Hence, the AUC value equals $(3 \times 1 + 2 \times 0.5)/6 \approx 0.67$. For precision, if $L = 2$, the predicted links are (1, 4) and (4, 5). Clearly, the former is wrong while the latter is right, and thus the precision equals 0.5.

3. Similarity-based algorithms

The simplest framework of link prediction methods is the similarity-based algorithm, where each pair of nodes, x and y, is assigned a score s_{xy}, which is directly defined as the similarity (or called proximity in the literature) between x and y. All non-observed links are ranked according to their scores, and the links connecting more similar nodes are supposed to be of higher existence likelihoods. In despite of its simplicity, the study on similarity-based algorithms is the mainstream issue. In fact, the definition of node similarity is a nontrivial challenge. Similarity index can be very simple or very complicated and it may work well for some networks while fails for some others. In addition, the similarities can be used in a more skilled way, such as being locally integrated under the *collaborative filtering*[5] framework [34].

Node similarity can be defined by using the essential attributes of nodes: two nodes are considered to be similar if they have many common features [35]. However, the attributes of nodes are generally hidden, and thus we focus on another group of similarity indices, called *structural similarity*, which is based solely on the network structure. The structural similarity indices can be classified in various ways, such as local vs. global, parameter-free vs. parameter-dependent, node-dependent vs. path-dependent, and so on. The similarity indices can also be sophisticatedly classified as *structural equivalence* and *regular equivalence*. The former embodies a latent assumption that the link itself indicated a similarity between two endpoints (see, for example, the *Leicht–Holme–Newman index* [36] and *transferring similarity* [37]), while the latter assumes that two nodes are similar if their neighbors are similar. Readers are encouraged to see Ref. [38] for the mathematical definition of regular equivalence and Ref. [39] for a recent application on the prediction of protein functions.

Here we adopt the simplest method, where 20 similarity indices are classified into three categories: the former 10 are local indices, followed by 7 global indices, and the last 3 are quasi-local indices, which do not require global topological information but make use of more information than local indices.

3.1. Local similarity indices

(1) *Common Neighbors* (CN). For a node x, let $\Gamma(x)$ denote the set of neighbors of x. In common sense, two nodes, x and y, are more likely to have a link if they have many common neighbors. The simplest measure of this neighborhood overlap is the directed count, namely

$$s_{xy}^{\mathrm{CN}} = |\Gamma(x) \cap \Gamma(y)|, \tag{2}$$

where $|Q|$ is the cardinality of the set Q. 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 [40] used this quantity in the study of collaboration networks, showing a positive correlation between the number of common neighbors and the probability that two scientists will collaborate in the future. Kossinets and Watts [14] analyzed a large-scale social network, suggesting that two students having many mutual friends are very probable to be friends in future. The following six indices are also based on the number of common neighbors, yet with different normalization methods.

(2) *Salton Index* [6]. It is defined as

$$s_{xy}^{\mathrm{Salton}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{k_x \times k_y}}, \tag{3}$$

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

(3) *Jaccard Index* [41]. This index was proposed by Jaccard over a hundred years ago, and is defined as

$$s_{xy}^{\mathrm{Jaccard}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}. \tag{4}$$

(4) *Sørensen Index* [42]. This index is used mainly for ecological community data, and is defined as

$$s_{xy}^{\mathit{Sørensen}} = \frac{2|\Gamma(x) \cap \Gamma(y)|}{k_x + k_y}. \tag{5}$$

[5] Collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. [33].

(5) *Hub Promoted Index* (HPI) [43]. This index is proposed for quantifying the topological overlap of pairs of substrates in metabolic networks, and is defined as

$$s_{xy}^{\mathrm{HPI}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\min\{k_x, k_y\}}. \tag{6}$$

Under this measurement, the links adjacent to hubs are likely to be assigned high scores since the denominator is determined by the lower degree only.

(6) *Hub Depressed Index* (HDI). Analogously to the above index, we also consider a measurement with the opposite effect on hubs, defined as

$$s_{xy}^{\mathrm{HDI}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\max\{k_x, k_y\}}. \tag{7}$$

(7) *Leicht–Holme–Newman Index* (LHN1) [36]. This index assigns high similarity to node pairs that have many common neighbors compared not to the possible maximum, but to the expected number of such neighbors. It is defined as

$$s_{xy}^{\mathrm{LHN1}} = \frac{|\Gamma(x) \cap \Gamma(y)|}{k_x \times k_y}, \tag{8}$$

where the denominator, $k_x \times k_y$, is proportional to the expected number of common neighbors of nodes x and y in the configuration model [44]. We use the abbreviation LHN1 to distinguish this index to another index (named as LHN2 index) also proposed by Leicht, Holme and Newman.

(8) *Preferential Attachment Index* (PA). The mechanism of preferential attachment can be 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 [45]. A similar mechanism can also lead to scale-free networks without growth [46], where at each time step, an old link is removed and a new link is generated. The probability that this new link will connect x and y is proportional to $k_x \times k_y$. Motivated by this mechanism, the corresponding similarity index can be defined as

$$s_{xy}^{\mathrm{PA}} = k_x \times k_y, \tag{9}$$

which has been widely used to quantify the functional significance of links subject to various network-based dynamics, such as percolation [47], synchronization [48] and transportation [49]. Note that, this index does not require the information of the neighborhood of each node, as a consequence, it has the least computational complexity.

(9) *Adamic–Adar Index* (AA) [50]. This index refines the simple counting of common neighbors by assigning the less-connected neighbors more weights, and is defined as

$$s_{xy}^{\mathrm{AA}} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k_z}. \tag{10}$$

(10) *Resource Allocation Index* (RA) [51]. This index is motivated by the resource allocation dynamics on complex networks [52]. Consider a pair of nodes, x and y, which are not directly connected. The node x can send some resource to y, with their common neighbors playing the role of transmitters. In the simplest case, we assume that each transmitter has a unit of resource, and will equally distribute it to all its neighbors. The similarity between x and y can be defined as the amount of resource y received from x, which is

$$s_{xy}^{\mathrm{RA}} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{k_z}. \tag{11}$$

Clearly, this measure is symmetric, namely $s_{xy} = s_{yx}$. Note that, although resulting from different motivations, the AA index and RA index have very similar form. Indeed, they both depress the contribution of the high-degree common neighbors. AA index takes the form $(\log k_z)^{-1}$ while RA index takes the form k_z^{-1}. The difference is insignificant when the degree, k_z, is small, while it is considerable when k_z is large. In other words, RA index punishes the high-degree common neighbors more heavily than AA.

Liben-Nowell et al. [58] and Zhou et al. [51] systematically compared a number of local similarity indices on many real networks: the former [58] focuses on social collaboration networks and the latter [51] considers disparate networks including the protein–protein interaction network, electronic grid, Internet, US airport network, etc. According to extensive experimental results on real networks (see results in Table 1), the RA index performs best, while AA and CN indices have the second best overall performance among all the above-mentioned local indices.

The PA index has the worst overall performance, yet we are interested in it for it requires the least information. Note that, PA performs even worst than pure chance for the Internet at router level and the power grid. In these two networks, the nodes have well-defined positions and the links are physical lines. Actually, geography plays a significant role and links with very long geographical distances are rare. As local centers, the high-degree nodes have longer geographical distances to each other than average, and thus have a lower probability of directly connecting to each other, which leads to the bad

Table 1
Accuracies of different local similarity indices subject to link prediction, measured by the AUC value. Each number is obtained by averaging over 10 implementations with independently random partitions of testing set (90%) and probe set (10%). The entries corresponding to the highest accuracies among these 10 indices are emphasized in black. The six real networks for testing are a protein–protein interaction network (PPI) [16], a co-authorship network of scientists who are themselves publishing on the topic of network science (NS) [53], an electrical power grid of the western US (Grid) [54], a network of the US political blogs (PB) [55], a router-level Internet collected by *Rocketfuel Project* (INT) [56], and a network of the US air transportation system (USAir) [57]. Detailed information about these networks can be found in Ref. [51].

Indices	PPI	NS	Grid	PB	INT	USAir
CN	0.889	**0.933**	**0.590**	0.925	**0.559**	0.937
Salton	0.869	0.911	0.585	0.874	0.552	0.898
Jaccard	0.888	**0.933**	**0.590**	0.882	**0.559**	0.901
Sørensen	0.888	**0.933**	**0.590**	0.881	**0.559**	0.902
HPI	0.868	0.911	0.585	0.852	0.552	0.857
HDI	0.888	**0.933**	**0.590**	0.877	**0.559**	0.895
LHN1	0.866	0.911	0.585	0.772	0.552	0.758
PA	0.828	0.623	0.446	0.907	0.464	0.886
AA	0.888	0.932	**0.590**	0.922	**0.559**	0.925
RA	**0.890**	**0.933**	**0.590**	**0.931**	**0.559**	**0.955**

performance of PA. In contrast, although USAir has well-defined geographical positions of nodes, its links are not physical. Empirical data has demonstrated that the number of airline flights is not very sensitive to the geographical distance within a big range [59,60] (another topological evidence for the relatively good performance of PA on USAir is the so-called rich-club phenomenon [61,62]). The LHN1 index performs the second worst, however, compared with all other neighborhood-based indices, it is very good at uncovering the missing links connecting two small-degree nodes [27].

Recently, Pan et al. [63] have compared all the local indices appeared in Ref. [51] in a similarity-based community detection algorithm, and their experimental results again indicate that the RA index performs best. Wang et al. [64] have applied the RA index to estimate the weights between stations in Chinese railway, which shows better performance than the CN index. In addition, the RA index for bipartite networks can be applied in personalized recommendation with higher accuracy than the classical collaborative filtering [65].

3.2. Global similarity indices

(11) *Katz Index* [66]. This index is based on the ensemble of all paths, which directly sums over the collection of paths and is exponentially damped by length to give the shorter paths more weights. The mathematical expression reads

$$s_{xy}^{\mathrm{Katz}} = \sum_{l=1}^{\infty} \beta^l \cdot |\mathrm{paths}_{xy}^{\langle l \rangle}| = \beta A_{xy} + \beta^2 (A^2)_{xy} + \beta^3 (A^3)_{xy} + \cdots, \tag{12}$$

where $\mathrm{paths}_{xy}^{\langle l \rangle}$ is the set of all paths with length l connecting x and y, and β is a free parameter (i.e., the damping factor) controlling the path weights. Obviously, a very small β yields a measurement close to CN, because the long paths contribute very little. The similarity matrix can be written as

$$S^{\mathrm{Katz}} = (I - \beta A)^{-1} - I. \tag{13}$$

Note that, β must be lower than the reciprocal of the largest eigenvalue of matrix A to ensure the convergence of Eq. (12).

(12) *Leicht–Holme–Newman Index* (LHN2) [36]. This index is a variant of the Katz index. Based on the concept that two nodes are similar if their immediate neighbors are themselves similar, one obtains a self-consistent matrix formulation

$$S = \phi A S + \psi I = \psi (I - \phi A)^{-1} = \psi (I + \phi A + \phi^2 A^2 + \cdots), \tag{14}$$

where ϕ and ψ are free parameters controlling the balance between the two components of the similarity. Setting $\psi = 1$, it is very similar to the Katz index. Note that, $(A^l)_{xy}$ is equal to the number of paths of length l from x to y. The expected value of $(A^l)_{xy}$, namely $E[(A^l)_{xy}]$, equals $(k_x k_y / 2M)\lambda_1^{l-1}$, where λ_1 is the largest eigenvalue of A and M is the total number of edges in the network. Replace $(A^l)_{xy}$ in Eq. (14) with $(A^l)_{xy}/E[(A^l)_{xy}]$, we obtain the expression:

$$s_{xy}^{\mathrm{LHN2}} = \delta_{xy} + \frac{2M}{k_x k_y} \sum_{l=0}^{\infty} \phi^l \lambda^{1-l} (A^l)_{xy} = \left[1 - \frac{2M\lambda_1}{k_x k_y}\right] \delta_{xy} + \frac{2M\lambda_1}{k_x k_y} \left[\left(I - \frac{\phi}{\lambda_1} A\right)^{-1}\right]_{xy}, \tag{15}$$

where δ_{xy} is the *Kronecker function*. Since the first item is a diagonal matrix, it can be dropped and thus we arrive to a compact expression

$$S = 2m\lambda_1 D^{-1} \left(I - \frac{\phi A}{\lambda_1}\right)^{-1} D^{-1}, \tag{16}$$

where D is the degree matrix with $D_{xy} = \delta_{xy}k_x$ and $\phi(0 < \phi < 1)$ is a free parameter. The choosing of ϕ depends on the investigated network, and smaller ϕ assigns more weights on shorter paths. Similar ideas can also be found in Ref. [67].

(13) *Average Commute Time* (ACT). Denote by $m(x, y)$ the average number of steps required by a random walker starting from node x to reach node y, the average commute time between x and y is

$$n(x, y) = m(x, y) + m(y, x), \tag{17}$$

which can be obtained in terms of the pseudoinverse[6] of the Laplacian matrix, L^+ ($L = D - A$), as Refs. [70,71]:

$$n(x, y) = M(l^+_{xx} + l^+_{yy} - 2l^+_{xy}), \tag{18}$$

where l^+_{xy} denotes the corresponding entry in L^+. Assuming two nodes are more similar if they have a smaller average commute time, then the similarity between the nodes x and y can be defined as the reciprocal of $n(x, y)$, namely (the constant factor M is removed)

$$s^{\mathrm{ACT}}_{xy} = \frac{1}{l^+_{xx} + l^+_{yy} - 2l^+_{xy}}. \tag{19}$$

(14) *Cosine based on* L^+. This index is an inner-product-based measure. In the Euclidean space spanned by $v_x = \Lambda^{\frac{1}{2}}U^T\vec{e}_x$, where U is an orthonormal matrix made of the eigenvectors of L^+ ordered in decreasing order of corresponding eigenvalue λ_x, $\Lambda = \mathrm{diag}(\lambda_x)$, $\vec{e}_x$ is an $N \times 1$ vector with the xth element equal to 1 and others all equal to 0, and T is the matrix transposition, the pseudoinverse of the Laplacian matrix are the inner products of the node vectors, $l^+_{xy} = v^T_x v_y$. Accordingly, the cosine similarity is defined as the cosine of the node vectors, namely [71]

$$s^{\cos^+}_{xy} = \cos(x, y)^+ = \frac{v^T_x v_y}{|v_x| \cdot |v_y|} = \frac{l^+_{xy}}{\sqrt{l^+_{xx} \cdot l^+_{yy}}}. \tag{20}$$

(15) *Random Walk with Restart* (RWR). This index is a direct application of the PageRank algorithm [72]. Consider a random walker starting from node x, who will iteratively moves to a random neighbor with probability c and return to node x with probability $1 - c$. Denote by q_{xy} the probability this random walker locates at node y in the steady state, we have

$$\vec{q}_x = cP^T\vec{q}_x + (1 - c)\vec{e}_x, \tag{21}$$

where P is the transition matrix with $P_{xy} = 1/k_x$ if x and y are connected, and $P_{xy} = 0$ otherwise. The solution is straightforward, as

$$\vec{q}_x = (1 - c)(I - cP^T)^{-1}\vec{e}_x. \tag{22}$$

The RWR index is thus defined as

$$s^{\mathrm{RWR}}_{xy} = q_{xy} + q_{yx}, \tag{23}$$

where q_{xy} is the yth element of the vector $\vec{q}_x$. A fast algorithm to calculate this index was proposed by Tong et al. [73], and the application of this index to recommender systems can be found in Ref. [74].

(16) *SimRank* [75]. Similar to the LHN2, SimRank is defined in a self-consistent way, according to the assumption that two nodes are similar if they are connected to similar nodes.

$$s^{\mathrm{SimRank}}_{xy} = C \cdot \frac{\sum_{z\in\Gamma(x)}\sum_{z'\in\Gamma(y)} s^{\mathrm{SimRank}}_{zz'}}{k_x \cdot k_y} \tag{24}$$

where $s_{xx} = 1$ and $C \in [0, 1]$ is the decay factor. The SimRank can also be interpreted by the random walk process, that is, $s^{\mathrm{SimRank}}_{xy}$ measures how soon two random walkers, respectively starting from nodes x and y, are expected to meet at a certain node.

(17) *Matrix Forest Index* (MFI) [76]. This index is defined as

$$S = (I + L)^{-1}, \tag{25}$$

where the similarity between x and y can be understood as the ratio of the number of spanning rooted forests such that nodes x and y belong to the same tree rooted at x to all spanning rooted forests of the network (see details in Ref. [76]). A parameter-dependent variant of MFI is

$$S = (I + \alpha L)^{-1}, \quad \alpha > 0. \tag{26}$$

[6] The pseudoinverse of a matrix is a generalization of the inverse matrix. It is used to compute a 'best fit' (least squares) solution to a system of linear equations that lacks a unique solution or to find the minimum (Euclidean) norm solution to a system of linear equations with multiple solutions. The pseudoinverse is defined and is unique for all matrices whose entries are real or complex numbers. It can be computed using the singular value decomposition [68,69].

Table 2
Accuracies of the three path-dependent similarity indices, measured by AUC and precision. Here, only the main components of example networks are considered (see Ref. [78] for detailed information). Each number is obtained by averaging over 10 independent realizations. The entries corresponding to the highest accuracies are emphasized in black. For LP, Katz and LHN2 indices, the AUC values are corresponding to the optimal parameter which will be used to calculate their corresponding precision where we set $L = 100$. For USAir, the optimal value of ϵ is negative (see the explanation in Ref. [51]). LP* denotes the LP index with a fixed parameter $\epsilon = 0.01$ (for USAir $\epsilon = -0.01$). The very small difference between the optimal case and the case with $\epsilon = 0.01$ suggests that in the real application, one can directly set ϵ as a very small number, instead of finding out its optimum that may cost much time. This again supports our motivation that the essential advantage of the usage of the second-order neighborhood is to improve the distinguishability of the similarity scores.

AUC	PPI	NS	Grid	PB	INT	USAir
LP	0.970	**0.988**	0.697	**0.941**	0.943	**0.960**
LP*	0.970	**0.988**	0.697	0.939	0.941	0.959
Katz	**0.972**	**0.988**	**0.952**	0.936	**0.975**	0.956
LHN2	0.968	0.986	0.947	0.769	0.959	0.778
Precision	PPI	NS	Grid	PB	INT	USAir
LP	**0.734**	**0.292**	**0.132**	**0.519**	**0.557**	**0.627**
LP*	**0.734**	**0.292**	**0.132**	0.469	0.121	**0.627**
Katz	0.719	0.290	0.063	0.456	0.368	0.623
LHN2	0	0.060	0.005	0	0	0.005

This index has been applied to quantify the similarity between nodes on collaborative recommendation task [77]. The results indicate that a simple nearest-neighbors rule based on similarity measured by MFI performs best.

Comparing with the local similarity indices, the global ones ask for the whole topological information. Although the global indices can provide much more accurate prediction than the local ones, they suffer two big disadvantages: (i) the calculation of a global index is very time consuming, and is usually infeasible for large-scale networks; (ii) sometimes, the global topological information is not available, especially if we would like to implement the algorithm in a decentralized manner. As we will show in the next subsection, a promising tradeoff is the quasi-local indices, which consider more information than local indices while abandon the superfluous information that makes no contribution or very little contribution to the prediction accuracy.

3.3. Quasi-local indices

(18) *Local Path Index* (LP) [51,78]. To provide a good tradeoff of accuracy and computational complexity, we here introduce an index that takes consideration of local paths, with wider horizon than CN. It is defined as

$$S^{\mathrm{LP}} = A^2 + \epsilon A^3, \tag{27}$$

where ϵ is a free parameter. Clearly, this measure degenerates to CN when $\epsilon = 0$. And if x and y are not directly connected (this is the case we are interested in), $(A^3)_{xy}$ is equal to the number of different paths with length 3 connecting x and y. This index can be extended to account for higher-order paths, as

$$S^{\mathrm{LP}(n)} = A^2 + \epsilon A^3 + \epsilon^2 A^4 + \cdots + \epsilon^{n-2} A^n, \tag{28}$$

where $n > 2$ is the maximal order. With the increasing of n, this index asks for more information and computation. Especially, when $n \to \infty$, $S^{\mathrm{LP}(n)}$ will be equivalent to the Katz index that takes into account all paths in the network. The computational complexity of this index in an uncorrelated network is $\mathbb{O}(N\langle k\rangle^n)$, which grows fast with the increasing of n and will exceed the complexity for calculating the Katz index (approximate to $\mathbb{O}(N^3)$) for large n. Experimental results show that the optimal n is positively correlated with the average shortest distance of the network [78].

The LP index performs remarkably better than the neighborhood-based indices, such as RA, AA and CN [51]. It is because the neighborhood information is less distinguishable and two node pairs are of high probability to be assigned the same similarity scores. Taking INT as an example, there are more than 10^7 node pairs, 99.59% of which are assigned zero score by CN. For all the node pairs having scores higher than 0, 91.11% are assigned score 1, and 4.48% are assigned score 2. Using a little bit more information involving the next nearest neighbors may break the "degeneracy of the states" and make the similarity scores more distinguishable. This is the reason why the LP index largely improves the prediction accuracy.

The comparison of LP index with other two path-dependent global indices, the Katz and LHN2 indices, is shown in Table 2. Overall speaking, the Katz index performs best subject to the AUC value, while the LP index is the best for the precision. For the network with small average shortest distance (e.g., USAir and PB), LP index gives the most accurate predictions for both AUC and precision. In a word, the LP index provides competitively good predictions while asks for much lighter computation compared with the global indices.

(19) *Local Random Walk* (LRW) [79]. 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

$$s_{xy}^{\mathrm{LRW}}(t) = q_x\pi_{xy}(t) + q_y\pi_{yx}(t). \tag{29}$$

Table 3
Comparison of algorithms' accuracy quantified by AUC and precision. For each network, the training set contains 90% of the known links. Each number is obtained by averaging over 1000 implementations with independently random divisions of training set and probe set. The parameters $\varepsilon = 10^{-3}$ for LP (for USAir, $\varepsilon = -10^{-3}$) and $c = 0.9$ for RWR. The numbers inside the brackets denote the optimal step of LRW and SRW indices. For example, 0.972(2) means the optimal AUC is obtained at the second step of LRW. The highest accuracy in each line is emphasized in black. For HSM, 5000 samples of dendrograms for each implementation are generated.

AUC	CN	RA	LP	ACT	RWR	HSM	LRW	SRW
USAir	0.954	0.972	0.952	0.901	0.977	0.904	0.972(2)	**0.978**(3)
NetScience	0.978	0.983	0.986	0.934	**0.993**	0.930	0.989(4)	0.992(3)
Power	0.626	0.626	0.697	0.895	0.760	0.503	0.953(16)	**0.963**(16)
Yeast	0.915	0.916	0.970	0.900	0.978	0.672	0.974(7)	**0.980**(8)
C.elegans	0.849	0.871	0.867	0.747	0.889	0.808	0.899(3)	**0.906**(3)
Precision	CN	RA	LP	ACT	RWR	HSM	LRW	SRW
USAir	0.59	0.64	0.61	0.49	0.65	0.28	0.64(3)	**0.67**(3)
NetScience	0.26	0.54	0.30	0.19	**0.55**	0.25	0.54(2)	0.54(2)
Power	0.11	0.08	**0.13**	0.08	0.09	0.00	0.08(2)	0.11(3)
Yeast	0.67	0.49	0.68	0.57	0.52	0.84	**0.86**(3)	0.73(9)
C.elegans	0.12	0.13	**0.14**	0.07	0.13	0.08	**0.14**(3)	**0.14**(3)

where q is the initial configuration function. In Ref. [79] Liu and Lü applied a simple form determined by node degree, namely $q_x = \frac{k_x}{M}$. Note that, here we only focus on the few-step random walk instead of the stationary state where we have $\pi_{xy} = \frac{k_y}{M}$ and thus leading to a local index.

(20) *Superposed Random Walk* (SRW) [79]. Similar to the RWR index, Liu and Lü [79] proposed the SRW index, where the random walker is continuously released at the starting point, resulting in a higher similarity between the target node and the nodes nearby. The mathematical expression reads

$$s_{xy}^{\mathrm{SRW}}(t) = \sum_{\tau=1}^{t} s_{xy}^{\mathrm{LRW}}(\tau) = \sum_{\tau=1}^{t} [q_x \pi_{xy}(\tau) + q_y \pi_{yx}(\tau)], \tag{30}$$

where t denotes the time steps.

Liu and Lü [79] systematically compared these two indices, LRW and SRW, with five other indices, including three local (or quasi-local) indices, CN, RA and LP, and two other random-walk-based global indices, ACT and RWR, as well as the *hierarchical structure method* (HSM) proposed by Clauset et al. [80] (see Section 4.1 for the detailed introduction of HSM). According to the experimental results (see Table 3), LRW and SRW methods perform better than other indices with their respective optimal walking step positively correlated with the average shortest distance of the network.

Furthermore, the computational complexity of LRW and SRW is lower than ACT and RWR whose time complexity in calculating inverse and pseudoinverse is approximately $\mathbb{O}(N^3)$, while the time complexity of n-steps LRW and SRW are approximately $\mathbb{O}(N\langle k\rangle^n)$, ignoring degree heterogeneity of the network. That is to say, when n is small LRW and SRW run much faster than other random-walk-based global similarity indices. The advantage of LRW and SRW for their low computational complexity is prominent especially in the huge size (i.e. large N) and sparse (i.e. small $\langle k\rangle$) networks. For example, LRW or SRW for power grid is thousand times faster than ACT, $\cos^+$ and RWR, even for $n \simeq 10$ [79].

With the similar motivation of LRW and SRW, Mantrach et al. recently proposed a bounded normalized random walk with restart algorithm (see Eq. (21) for the definition of RWR), and applied it to address the classification problem [81]. With this method both complexities of time and space can be reduced.

4. Maximum likelihood methods

This section will introduce two recently proposed algorithms based on the maximum likelihood estimation. These algorithms presuppose some organizing principles of the network structure, with the detailed rules and specific parameters obtained by maximizing the likelihood of the observed structure. Then, the likelihood of any non-observed link can be calculated according to those rules and parameters.

From the viewpoint of practical applications, an obvious drawback of the maximum likelihood methods is that it is very time consuming. A well designed algorithm is able to handle networks with up to a few thousand nodes in a reasonable time, but will definitely fail to deal with the huge online networks that often consist of millions of nodes. In addition, the maximum likelihood methods are probably not among the most accurate ones (see, for example, the comparison between hierarchical structure model and some typical similarity-based methods in Table 3). However, the maximum likelihood methods provide very valuable insights into the network organization, which cannot be gained from the similarity-based algorithms or the probabilistic models.

4.1. Hierarchical structure model

Empirical evidence indicates that many real networks are hierarchically organized, where nodes can be divided into groups, further subdivided into groups of groups, and so forth over multiple scales [21] (e.g., metabolic networks [43] and

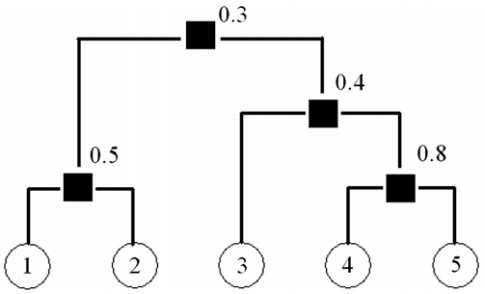

Fig. 2. Illustration of a dendrogram of a network with 5 nodes. Accordingly, the connecting probability of nodes 1 and 2 is 0.5, of nodes 1 and 3 is 0.3, of nodes 3 and 4 is 0.4.

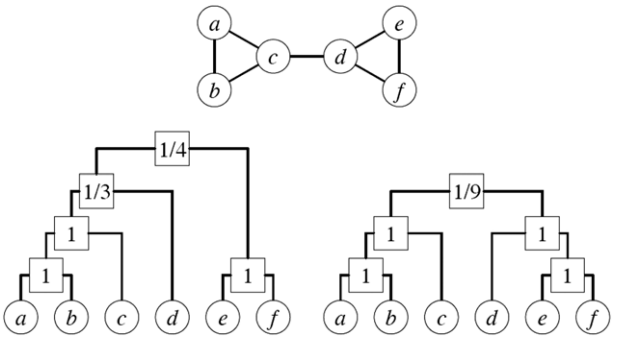

Fig. 3. The likelihood of two possible dendrograms for an example network consisting of 6 nodes. The interval nodes are labeled with the maximum likelihood probability obtained by Eq. (32). The likelihoods are $\mathcal{L}(D_1) \approx 0.00165$ (left dendrogram) and $\mathcal{L}(D_1) \approx 0.0433$ (right dendrogram). Copyright is held by Nature Publishing Group.
Source: Reprinted figure with permission from Ref. [80].

brain networks [82]). As Redner said [83], focusing on the hierarchical structure inherent in social and biological networks might provide a smart way to find missing links. Clauset et al. [80] proposed a general technique to infer the hierarchical organization from network data and further applied it to predict the missing links.

As shown in Fig. 2, the hierarchical structure of a network can be represented by a dendrogram with N leaves (corresponding to the nodes of the network) and $N-1$ internal nodes. Clauset et al. [80] introduced a simple model where each internal node r is associated with a probability p_r and the connecting probability of a pair of nodes (leaves) is equal to $p_{r'}$ where r' is the lowest common ancestor of these two nodes. Given a real network G and a dendrogram D, let E_r be the number of edges in G whose endpoints have r as their lowest common ancestor in D, and let L_r and R_r, respectively, be the number of leaves in the left and right subtrees rooted at r. Then the likelihood of the dendrogram D together with a set of p_r is

$$\mathcal{L}(D, \{p_r\}) = \prod_r p_r^{E_r} (1 - p_r)^{L_r R_r - E_r}. \tag{31}$$

For a fixed D, it is obvious that

$$p_r^* = \frac{E_r}{L_r R_r} \tag{32}$$

maximizes $\mathcal{L}(D, \{p_r\})$. Therefore, according to the *maximum likelihood method* [84], with a fixed D, it is easy to determine $\{p_r\}$ (by Eq. (32)) that best fits the network G. Fig. 3 shows an example network and two possible dendrograms, as well as the corresponding likelihoods. It is in accordance with the common sense that D_2 is more likely. The *Markov chain Monte Carlo method* is used to sample dendrograms with probability proportional to their likelihood (see the Supplementary Information of Ref. [80] and a benchmark book [85] for details).

The algorithm to predict the missing links contains the following procedures: (i) sample a large number of dendrograms with probability proportional to their likelihood; (ii) for each pair of unconnected nodes i and j, calculate the mean connecting probability $\langle p_{ij} \rangle$ by averaging the corresponding probability p_{ij} over all sampled dendrograms; (iii) sort these node pairs in descending order of $\langle p_{ij} \rangle$ and the highest-ranked ones are those to be predicted. According to the AUC value, this algorithm outperforms the CN index for the terrorist association network [86] and the grassland species food web [87], while loses for the metabolic network of the spirochaete *Treponema Pallidum* [88].

The hierarchical structure model provides a smart way to predict missing links, and, maybe more significantly, it uncovers the hidden hierarchical organization of networks. However, as mentioned above, a big disadvantage is that this algorithm

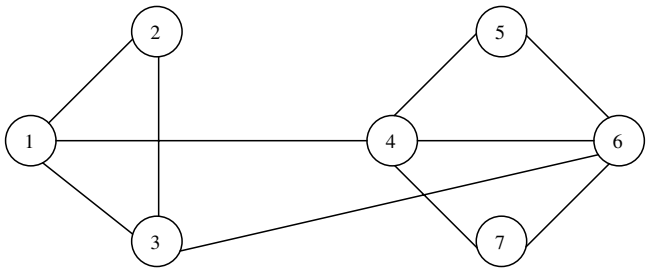

Fig. 4. An illustration about the calculation of likelihood for the stochastic block model.

runs very slow. Actually, the process to sample dendrograms usually asks for $\mathbb{O}(N^2)$ steps of the Markov chain [80], and in the worst case, it takes exponential time [89]. In comparison, according to the CPU of an advanced desktop computer, the hierarchical structure model cannot manage a network of tens of thousand nodes, while the algorithms based on local similarity indices can deal with networks with tens of million nodes. Another noticeable remark is that this model may give poor predictions for those networks without clear hierarchical structures.

4.2. Stochastic block model

Stochastic block model [90–93] is one of the most general network models, where nodes are partitioned into groups and the probability that two nodes are connected depends solely on the groups to which they belong. The stochastic block model can capture the community structure [22], role-to-role connections [94,95] and maybe other factors for the establishing of connections, especially when the group membership plays a considerable role in determining how nodes interact with each other, which usually could not be well described by the simple assortativity coefficient [96,97] or the degree–degree correlations [98,99].

Given a partition $\mathcal{M}$ where each node belongs to one group and the connecting probability for two nodes respectively in groups α and β is denoted by $Q_{\alpha\beta}$ ($Q_{\alpha\alpha}$ represents the probability that two nodes within group α are connected), then the likelihood of the observed network structure is [18]

$$\mathcal{L}(A|\mathcal{M}) = \prod_{\alpha\leq\beta} Q_{\alpha\beta}^{l_{\alpha\beta}}(1-Q_{\alpha\beta})^{r_{\alpha\beta}-l_{\alpha\beta}}, \tag{33}$$

where $l_{\alpha\beta}$ is the number of edges between nodes in groups α and β and $r_{\alpha\beta}$ is the number of pairs of nodes such that one node is in α and the other is in β. Similar to Eq. (32), the optimal $Q_{\alpha\beta}$ that maximizes the likelihood $\mathcal{L}(A|\mathcal{M})$ is

$$Q^*_{\alpha\beta} = \frac{l_{\alpha\beta}}{r_{\alpha\beta}}. \tag{34}$$

A simple illustration is shown in Fig. 4. Given a partition $\mathcal{M} = \{\{1, 2, 3\}, \{4, 5, 6, 7\}\}$, according to Eq. (34), the Q values corresponding to the maximum likelihood are $Q^*_{11} = \frac{3}{3} = 1$, $Q^*_{12} = \frac{2}{12} = \frac{1}{6}$, $Q^*_{22} = \frac{5}{6}$, and thus the likelihood is

$$\mathcal{L} = 1 \times \left(\frac{1}{6}\right)^2 \left(\frac{5}{6}\right)^{10} \times \left(\frac{5}{6}\right)^5 \left(\frac{1}{6}\right) \approx 3.005 \times 10^{-4}. \tag{35}$$

Denote by Ω the set of all possible partitions, using *Bayes' Theorem* [100], the *reliability* of an individual link is [18]

$$R_{xy} = \mathcal{L}(A_{xy} = 1|A) = \frac{\int_\Omega \mathcal{L}(A_{xy} = 1|\mathcal{M})\mathcal{L}(A|\mathcal{M})p(\mathcal{M})\mathrm{d}\mathcal{M}}{\int_\Omega \mathcal{L}(A|\mathcal{M}')p(\mathcal{M}')\mathrm{d}\mathcal{M}'}, \tag{36}$$

where $p(\mathcal{M})$ is a constant assuming no prior knowledge about the model. Note that, the number of different partitions of N elements grows faster than any finite power of N, and thus even for a small network, to sum over all partitions is not possible in practice. The *Metropolis algorithm* [101] can be applied to estimate the link reliability [18]. Even though, the whole process is very time consuming, this method can only manage networks with up to a few thousands of nodes.

Reliability describes the likelihood of the existence of a link (i.e., the probability that the link "truly" exists) given the observed structure [18], which can be used not only to predict missing links (the nonexistent links in the observed network yet with the highest reliabilities) but also to identify possible spurious links (the existent links with the lowest reliabilities). Empirical comparison on five disparate networks (the social interactions in a karate club [102], the social network of frequent associations between 62 dolphins [103], the air transportation network of Eastern Europe [104], the neural network of the nematode *Caenorhabditis elegans* [105], and the metabolic network of *Escherichia coli* [106]) indicated that the overall performance of the maximum likelihood method based on stochastic block model [18] is better than the one based on the hierarchical structure model [80] and the similarity-based algorithm for common neighbors [58].

5. Probabilistic models

Probabilistic models aim at abstracting the underlying structure from the observed network, and then predicting the missing links by using the learned model. Given a target network $G = (V, E)$, the probabilistic model will optimize a built target function to establish a model composed of a group of parameters Θ, which can best fit the observed data of the target network. Then the probability of the existence of a nonexistent link (i, j) is estimated by the conditional probability $P(A_{ij} = 1|\Theta)$. This section will introduce the three mainstream methods, respectively called *Probabilistic Relational Model* (PRM) [107], *Probabilistic Entity Relationship Model* (PERM) [108] and *Stochastic Relational Model* (SRM) [109]. Note that, in some literature, the term PRM only refers to a specific model which is usually called the *Relational Bayesian Networks* nowadays, while we adopt the more general usage of PRM in this review.

5.1. Probabilistic relational models

PRMs represent a joint probability distribution over the attributes of a relational dataset. They allow the properties of an object to depend probabilistically both on other properties of that object and on properties of the related objects. Different from the traditional graphical models using a single graph to model the relationship among the attributes of homogeneous entities, PRMs contain three graphs [110]: the data graph G_D, the model graph G_M, and the inference graph G_I. These correspond to the skeleton, model, and ground graph as outlined by Heckerman et al. [111].

The data graph $G_D = (V_D, E_D)$ presents the input network, where nodes are the objects in the data and edges represent the relationships among the objects. Each node $v_i \in V_D$ and edge $e_j \in E_D$ are associated with a type $T(v_i) = t_{v_i}$, $T(e_j) = t_{e_j}$. Each item (either object or edge) type $t \in T$ has a number of associated attributes X^t. Consequently, each object v_i and link e_j are associated with a set of attribute values, $\mathrm{x}_{v_i}^{t_{v_i}}$ and $\mathrm{x}_{e_j}^{t_{e_j}}$, determined by their types, t_v and t_e, respectively. A PRM represents a joint probability distribution over the values of all the attributes in the data graph, $\mathrm{x} = \{\mathrm{x}_{v_i}^{t_{v_i}} : v_i \in V_D, T(v_i) = t_{v_i}\} \bigcup \{\mathrm{x}_{e_j}^{t_{e_j}} : e_j \in E_D, T(e_j) = t_{e_j}\}$. For example, in the student–course selection system, the students and courses are nodes and the edges represent the *select* relationship between students and courses. Clearly, there are two types of nodes, namely student and course. And the type *student* has four attributes: grade, age, sex and department, while the type *course* has five attributes: category, teacher, year, time and discipline.

The model graph $G_M = (V_M, E_M)$ represents the dependencies among attributes at the level of item types. Attributes of an item can depend probabilistically on other attributes of the same item, as well as on attributes of other related objects or links in G_D. Each node in V_M corresponds to an attribute $X_i^t \in \mathrm{X}^t$ where $t \in T$. The attributes with the same type in G_D are tied together. Thus G_D is decomposed into multiple examples of each type, based on which a joint model of dependencies among the type attributes can be built. G_M contains two parts: the dependent structure among all the type attributes and the conditional probability distribution (CPD) associated with the nodes in G_M.

The inference graph $G_I = (V_I, E_I)$ represents the probabilistic dependencies among all the variables in a single test set. It can be instantiated by a roll-out process of G_D and G_M. Each item-attribute pair in G_D gets a separate, local copy of the corresponding CPD from G_M. The relations in G_D determine the way that G_M is rolled out to form G_I. Therefore the structure of G_I is determined by both G_D and G_M.

With respect to different representations of the modeled graph G_M and the corresponding learning and inferring procedures, PRMs can be classified into three groups: Relational Bayesian Networks (RBNs), Relational Markov Networks (RMNs) and Relational Dependency Networks (RDNs).

Relational Bayesian Networks (RBNs) [107,112]: The model graph presented by a RBN is a *directed acyclic graph*[7] with a set of CPDs, P, to represent a joint distribution over the attributes of the item types. The set P contains a conditional probability distribution for each variable given its parents,[8] $p(x|pa_x)$, where pa_x denotes the parents of node x. Thus the joint probabilistic distribution can be calculated as

$$p(\mathrm{x}) = \prod_{t \in T} \prod_{X_i^t \in \mathrm{X}^t} \prod_{v:T(v)=t} p(x_{v_i}^t | pa_{x_{v_i}^t}) \prod_{e:T(e)=t} p(x_{e_i}^t | pa_{x_{e_i}^t}). \tag{37}$$

The need to avoid cycles in RBN leads to significant representational and computational difficulties. Inference is done by creating the complete ground network, which limits their scalability. RBN requires specifying a complete conditional model for each attribute of each class, which in large complex domains can be quite burdensome.

Relational Markov Networks (RMNs) [113,114]: A RMN uses an undirected graph and a set of potential functions Φ to represent the joint distribution over the attributes of the item types. Denote by C the set of cliques in the graph and each clique $c \in C$ is associated with a set of variables X_c ($\in \mathrm{X}^t$) (i.e. the nodes in this clique[9]) and a clique potential $\Phi_c(x_c)$

[7] A directed graph is acyclic if there is no directed path that starts and ends at the same variable. This constrain indicates that a random variable does not depend, directly or indirectly, on its own value.

[8] A direct link from a to b indicates that a is b's parent node.

[9] Actually a node in a clique corresponds to an attribute in the data graph.

which is a non-negative function over the possible values for $x_c \in \mathrm{X}_c$, then the joint probability over x is calculated with the formula

$$p(\mathrm{x}) = \frac{1}{Z}\prod_{c\in C}\Phi_c(x_c), \tag{38}$$

where Z is a normalizing constant, which sums over all possible instantiations. RMNs are trained discriminatively, and do not specify a complete joint distribution for the variables in the model. The learning algorithm uses maximum a posteriori (MAP) estimation with belief propagation for inference, which leads to a high computational complexity for learning.

Relational Dependency Networks (RDNs) [115,116]: The RDN is a bi-directed graphical model with a set of conditional probability distributions, which can be used to represent the cyclic dependencies in a relational setting. RDNs use pseudo-likelihood learning techniques to estimate an efficient approximation of the full joint distribution of the attribute values in a relational dataset. The pseudo-likelihood for data graph G_D is computed as a product over the item types t, the attributes of that type X^t, and the nodes v and edges e of that type.

$$PL(G_D;\Theta) = \prod_{t\in T}\prod_{X_i^t\in \mathrm{X}^t}\prod_{v:T(v)=t} p(x^t_{v_i}|pa_{x^t_{v_i}};\Theta)\prod_{e:T(e)=t} p(x^t_{e_i}|pa_{x^t_{e_i}};\Theta). \tag{39}$$

The CPDs in the RDN pseudo-likelihood are not required to factor the joint distribution of G_D. More specifically, when consider the variable $x^t_{v_i}$, we condition on the values of the parents $pa_{x^t_{v_i}}$ regardless of whether the estimation of CPDs for variables in $pa_{x^t_{v_i}}$ was conditioned on $x^t_{v_i}$. RDN adopts *Gibbs sampling*[10] to iteratively relabel each unobserved variable by drawing from its local conditional distribution, given the current state of the rest of the graph.

5.2. Probabilistic entity-relationship models

A specific type of probabilistic entity-relationship model is the directed acyclic PERM (DAPER for short), which uses directed arcs[11] to describe the relationship between attributes [108]. DAPER makes relationships the first class objects in the modeling language, and encourages an explicit representation of conditional probabilistic distribution. The DAPER model consists of six classes [118]. (i) Entity classes: specify the classes of objects in real world. (ii) Relationship classes: represent the interactions among entity classes. (iii) Attribute classes: describe properties of entities or relationships. (iv) Arc classes: represent probabilistic dependencies among corresponding attributes. (v) Local distribution classes: construct the local distributions for attributes corresponding to the attribute class. (vi) Constraint classes: specify how to derive inference graph (i.e., ground graph) from the corresponding DAPER model over the given instantiated domain. DAPER model assigns relationships the same importance as the entities.

The DAPER model can be used in situations where the relational structure itself is uncertain. And it is more expressive than either PRMs or *plate models*.[12] Actually, DAPER combines the features of plate models and PRMs, and the relations between DAPER models, PRMs and plate models can be found in Ref. [108].

5.3. Stochastic relational models

The key idea of SRM is to model the stochastic structure of entity relationships (i.e., links) via a tensor interaction of multiple Gaussian Processes (GPs), each defined on one type of entities [109].

Assuming that the observable links r are derived as local measurements of a real-value latent relation function t: $\mathcal{U}\times\mathcal{V}\to\mathbb{R}$, and each link r_{ij} is solely dependent on its latent value t_{ij}, modeled by the likelihood $p(r_{ij}|t_{ij})$. The latent relation function t is generated via a tensor interaction of two independent entity-specific GPs, one acting on $\mathcal{U}$ and the other on $\mathcal{V}$. Note that, $\mathcal{U}$ and $\mathcal{V}$ can both encompass an infinite number of entities. The relational processes are characterized by hyper-parameters $\Theta = \{\Theta_\Sigma, \Theta_\Omega\}$, where Θ_Σ and Θ_Ω are for the GP kernel function on $\mathcal{U}$ and $\mathcal{V}$ respectively. Thus SRM defines a Bayesian prior $p(t|\Theta)$ for the latent variables t. Let $\mathbb{I}$ be the index set of entity pairs having observed links, the marginal likelihood under such a prior is

$$p(\mathbf{R}_\mathbb{I}|\Theta) = \int\prod_{(ij)\in\mathbb{I}} p(r_{ij}|t_{ij})p(t|\Theta)\mathrm{d}t, \tag{40}$$

where $\mathbf{R}_\mathbb{I} = r_{ij}$, $(i,j)\in\mathbb{I}$. The hyper-parameters Θ can be estimated by maximizing the marginal likelihood. Then the link for a new pair of entities can be predicted by marginalizing over the *a posteriori* $p(t|\mathbf{R}_\mathbb{I},\Theta)$.

---

10 Heckerman et al. [115] proved that the Gibbs sampling can be used to estimate the joint distribution of a dependency network. For a basic introduction and summary of the Gibbs sampling, see Ref. [117].

11 In graph theory, the term "arc" stands for directed link.

12 The standard description of plate models can be found in Refs. [119,120]. Heckerman et al. [108] provided a new definition of plate model, which is slightly different from the traditional one [119,120]. According to this new definition, the plate models and DAPER models are equivalent, and a plate model can be invertibly mapped to a DAPER model [108].

This model in fact defines a set of non-parametric priors on infinite-dimensional tensor matrices, where each element represents a relationship between a tuple of entities. By maximizing the marginalized likelihood, information is exchanged between the participating GPs through the entire relational network, so that the dependency structure of links is messaged to the dependency of entities, reflected by the adapted GP kernels. Because the training is on a conditional model of links, this model offers a discriminative approach for link prediction, namely predicting the existences, strengths, or types of relationships based on the partially observed linkage network as well as the attributes of entities if given. Yu et al. further upgraded SRM with an edge-wise covariance with which the overall computational complexity can be reduced [121]. For more details one can see Refs. [109,122].

6. Applications

The problem of link prediction has attracted much attention from disparate research communities. This is mainly attributed to its broad applicability. For some networks, especially biological networks such as protein–protein interaction networks, metabolic networks and food webs, the discovery of links or interactions is costly in the laboratory or the field. A highly accurate prediction can reduce the experimental costs and speed the pace of uncovering the truth [80,83]. Link prediction has also been applied in the analysis of social networks, such as the prediction of being actors in acts [123], the prediction of the collaborations in co-authorship networks [58], the detection of the underground relationships between terrorists [80], and so on. In addition, the process of recommending items to users can be considered as a link prediction problem in the user-item bipartite networks [124,125]. Actually, almost the same techniques as the similarity-based link prediction has been applied in personalized recommendation [65,126–128]. Accurate recommendation can be used in e-commerce web sites to enhance the sales [129]. Moreover, the link prediction approaches can be applied to solve the classification problem in partially labeled networks, such as the prediction of protein functions [39], the detection of anomalous email [130], distinguishing the research areas of scientific publications [131], and finding out the fraud and legit users in cell phone networks [132]. The following three subsections will introduce typical applications of link prediction.

6.1. Reconstruction of networks

Guimerà and Sales-Pardo [18] considered the reconstruction of networks from the observed networks with missing and spurious links. Although one can rank the observed and non-observed links according to their reliabilities (see Eq. (36)), it is not easy to reconstruct the "true" network since generally no one knows how many missing and spurious links there are. Applying the similar techniques presented in Section 4.2, Guimerà and Sales-Pardo [18] defined the reliability of a network A as

$$R(A) = \prod_{A_{xy}=1, x<y} R_{xy} = \prod_{A_{xy}=1, x<y} \mathcal{L}(A_{xy} = 1|A^O), \tag{41}$$

where R_{xy} and $\mathcal{L}$ are defined in Eqs. (33) and (36), and the term A^O is used to emphasize that the likelihoods are calculated according to the observed network.

Given A^O, a straightforward idea is to find out the network A that maximizes the reliability defined by Eq. (41). However, the computation is too costly to be implemented. In practice, Guimerà and Sales-Pardo [18] designed a simple greedy algorithm. Their algorithm starts by evaluating the link reliabilities for all pairs of nodes. Then, at each time step it removes the link with the lowest reliability and adds the link (not yet in the current network) with the highest reliability. This change is accepted if and only if the network reliability increases. If it is rejected, the link with the next lowest reliability and the not-yet-existent link with the next highest reliability will be the next candidate for swapping. The algorithm stops if it rejects five consecutive attempts to swap links. The observed network is set as the initialization of the algorithm, and it will consecutively become another network with higher reliability than the initial network. Guimerà and Sales-Pardo [18] tested their algorithm by generating hypothetical observed networks A^O from the true networks A^T (the five true networks used for testing are introduced in Section 4.2). Each observation has a fraction of the true links removed and an identical number of random links added.

Guimerà and Sales-Pardo [18] compared the global network properties of the observed networks and those of the reconstructed networks. The reconstruction generally improves the estimates of clustering coefficient [54], modularity [133], assortativity [96,97], congestability,[13] synchronizability[14] and spreading threshold,[15] indicating the validity of the approach. Note that, the results from the greedy algorithm may be far different from the real optimum subject to the maximal reliability, thus we may expect even better estimates if one has developed a more effective and/or efficient algorithm. Readers should be warned that in both the algorithm and the preparation of observed networks, a latent assumption is that the number of missing links and the number of spurious links are equal. Since in the real systems, these two numbers may be very

[13] The congestability refers to the maximal betweenness centrality which governs the transportation throughput of a network [134,135].

[14] The synchronizability refers to the ratio between the largest and the smallest non-zero eigenvalues of the Laplacian matrix of a network, which quantifies the ability of synchronization under the framework of master stability analysis [136,137].

[15] Ignoring the degree–degree correlations and applying the mean-field approximation, the spreading threshold equals the ratio between the first and the second moments of the degree distribution [138,139].

different (it is easy to imagine that in many networks, such as metabolic networks and friendship networks, the missing links are much more than the spurious links), the effectiveness of the algorithm still needs further validation.

6.2. Evaluation of network evolving mechanisms

Since the groundbreaking work by Barabási and Albert [45], the evolving models all the time lie in the center of the complex network study. A fundamental difficulty is that for a given network or a target network property, there are generally many possible mechanisms and it is not easy to judge which one is the best. Taking the power-law degree distribution as an instance, the well-known mechanisms include *rich gets richer* [45], *good gets richer* [140], *optimal design* [141], *Hamiltonian dynamics* [142], *merging and regeneration* [143], *stability constraints* [144], and so on. Hence we cannot easily know which factor(s) leads to the scale-free property of a real network, and in fact there can be so many models competing for the final explanation of a given real network. It is very hard to evaluate different models by comparing their resulted networks with the target network, since there are too many metrics for topological features [5]. As mentioned in Section 1, there are many models about the topology of the Internet, some more accurately reproduce the degree distribution and the disassortative mixing pattern (e.g., see Ref. [19]) and some better characterize the k-core structure (e.g., see Ref. [20]). To judge which model (i.e., which evolving mechanism) is better than the others is a tough task.

Essentially speaking, an algorithm for link prediction makes a guess about the factors resulting in the existence of links, which is actually what an evolving model wants to show. In other words, an evolving model in principle can be mapped to a link prediction algorithm.[16] Therefore, we can quantitatively compare the accuracies of different evolving models with the help of the performance metrics for link prediction (see Section 2). We hope this methodology could provide a fair platform to compare different evolving models, which may be significant for the studies of network modeling. Next, we will show a real application about the Chinese city airline network, where each node represents a city with airport, and two cities are connected if there exists at least one direct airline between them [60].

It is well known that the evolution of a city airline network is affected by not only the topological factor, but also the geographical factor [145] and external factors, such as population and economic level of a city [146]. As shown by Liben-Nowell et al. [58] and Zhou et al. [51], the common neighbor index is a good candidate to account for the topological effects. In addition, Cui et al. [147] developed an evolving model driven completely by the common neighborhood, which well reproduces not only the global network properties, but also the local structural features like power-law clique-degree distributions [148] of social and technological networks. Therefore, we simply use the common neighbor index S^{CN} (see Eq. (2)) to represent the topological ingredient. Geographical distance is considered to be one of the realistic factors that affect the existence of nodes' interactions in networks [149]. Especially, it plays a very important role in analyzing transportation networks [150,151]. It is known to be relevant to the existence of an airline, and the number of airlines decays with the increasing of corresponding distance [59,145]. Accordingly, we use the inverse of geographical distance between two cities as the similarity index, say

$$s_{xy}^{\mathrm{DIS}} = \frac{1}{D_g(x, y)}, \tag{42}$$

where $D_g(x, y)$ denotes the geographical distance between cities x and y. Based on a null assumption that people in every city have the same frequency of air travels, the similar index for populations is defined as

$$s_{xy}^{\mathrm{POPU}} = P(x) \times P(y), \tag{43}$$

where $P(x)$ is the population of city x. The economic level of a city can be roughly quantified by its *gross domestic product* (GDP),[17] and thus the corresponding similarity is defined as

$$s_{xy}^{\mathrm{GDP}} = G(x) \times G(y), \tag{44}$$

where $G(x)$ denotes the GDP of city x. Considering that the airline business is most tightly related to the service industry, besides the simple GDP, we use the third sector of GDP, named the *tertiary industry*[18] to characterize a city's potential to build airlines.

$$s_{xy}^{\mathrm{TI}} = T(x) \times T(y), \tag{45}$$

where $T(x)$ is the tertiary industry of city x.

16 Note that, the evolving models consider not only the links between existed nodes but also the new connections involving the new added nodes. The structure-based similarity methods can only be applied to give predictions in the former case, while the methods utilizing some external factors may help in the latter case.

17 The GDP is a measure of a city's overall economic output. It is the market value of all final goods and services made within the borders of a city in a year. Here we use the data of the year 2005.

18 The tertiary industry (also called *tertiary sector of the economy*, *service sector* or *service industry*) consists of the "soft" parts of the economy, namely activities where people offer their knowledge and time to improve productivity, performance, potential, and sustainability. The basic characteristic of this sector is the production of services instead of end products.

Table 4
The prediction accuracy of the five similarity indices for the Chinese city airline network. The training and testing sets are divided according to the leave-one-out method.

Similarity indices	AUC
S^{CN}	0.898
S^{DIS}	0.699
S^{POPU}	0.745
S^{GDP}	0.855
S^{TI}	0.881

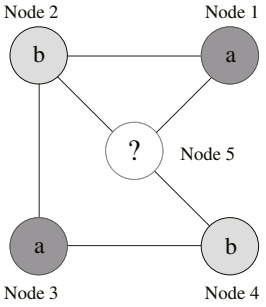

Fig. 5. An illustration of how to predict the fifth node's label by adding artificial links.

Since the size of the Chinese city airline network [60] is small ($|V| = 121$, $|E| = 1378$), we adopt the *leave-one-out* method, namely at each time, we pick only one link for test and all other links constitute the training set. This procedure repeats for 1378 times with each link being once the testing link. Table 4 displays the prediction accuracy (AUC values) of the five similarity indices. It indicates that every factor under consideration plays a role, while the topological factor is most significant. The tertiary industry of a city, as an external factor, also plays a very important role. Actually, a linear combination of the common neighbor index and the tertiary industry, as $S' = \lambda S^{\mathrm{CN}} + (1-\lambda)S^{\mathrm{TI}}$ can achieve a very high AUC value, 0.928, at $\lambda \approx 0.2$.

Although the method introduced here is straightforward, it gives insights into the underlying evolving mechanisms which may not be seriously considered in the early studies. The validity of this method has been demonstrated by some recent evidences. For example, by comparing the evolving models driven, respectively, by the topological factor, the geographical factor, and the above-mentioned three external factors, Liu et al. [152] showed that only the one considering the tertiary industry can reproduce the observed double power-law degree distribution of the Chinese city airline network. In addition, among many external factors, the *Granger causality test*[19] shows that the tertiary industry is the most significant factor in determining the passenger volume [154].

6.3. Classification of partially labeled networks

Given a network with partial nodes being labeled, the problem is to predict the labels of these unlabeled nodes based on the known labels and the network structure. Two main difficulties in achieving highly accurate classification are the sparsity of the known labeled nodes and the inconsistency of label information. To address these two difficulties, a simple but effective method is to add artificial connections between every pair of labeled and unlabeled nodes according to their similarity scores [131,155], with almost the same techniques used in similarity-based link prediction. An underlying assumption is that two nodes are more likely to be categorized into the same class if they are more similar to each other.

Consider an unweighted undirected network of both labeled and unlabeled nodes: $G(V, E, L)$, where V is the set of nodes, E is the set of links and $L = \{l_1, l_2, \ldots, l_m\}$ is the set of labels. The nodes without labels are labeled by 0. For each pair of nodes, x and y, a similarity index will assign a score as s_{xy}. For an unlabeled node x, the probability it belongs to l_i is

$$p(l_i|x) = \frac{\sum_{\{y|y\neq x,\,\mathrm{label}(y)=l_i\}} s_{xy}}{\sum_{\{y|y\neq x,\,\mathrm{label}(y)\neq 0\}} s_{xy}}, \tag{46}$$

[19] The Granger causality test is a technique for determining whether one time series is useful in forecasting another. See Ref. [153] for details.

where $l_i \in L$. The predicted label of node x is determined by the largest $p(l_i|x)$. If there are more than one maximum value, we randomly select one.

A simple example is shown in Fig. 5, where there are two kinds of labels (i.e. a and b) and five nodes, four of which are labeled already. Our task is to predict the label of the node 5. According to the common neighbors index S^{CN}, we obtain the similarity between node 5 and the other four labeled nodes: $s_{15} = 1$, $s_{25} = 1$, $s_{35} = 2$ and $s_{45} = 0$. Thus, the probabilities that node 5 belongs to classes a and b are $p(a|\mathtt{node5}) = 0.75$ and $p(b|\mathtt{node5}) = 0.25$, respectively. If we use RA index, the similarity scores are $s_{15} = \frac{1}{3}$, $s_{25} = \frac{1}{2}$, $s_{35} = \frac{1}{3} + \frac{1}{2}$ and $s_{45} = 0$. Therefore, the probabilities change to $p(a|\mathtt{node5}) = 0.7$ and $p(b|\mathtt{node5}) = 0.3$. According to any of the two indices, the predicted label of node 5 is a.

7. Outlook

In this article, we briefly summarized the progress of studies on link prediction, emphasizing on the recent contributions by statistical physicists. Although link prediction is not a new problem in information science, traditional methods have not caught up the new development of network science, especially the new perspectives and tools resulted from the studies of complex networks. In our opinion, the studies of link prediction and complex networks will benefit each other, because the in-depth understanding of network structure can be used to design advanced link prediction algorithms (e.g., making use of the information about hierarchical organization [80] and modular structure [18] of real networks to better predict missing links) and the performance of a link prediction algorithm could give evidences about structural features [51] as well as the algorithms themselves can be used to improve the estimates of real networks' properties [18] and to evaluate the evolving network models. In a word, to statistical physicists, the study of link prediction is just unfolding.

The significant contribution of the study of complex networks to the link prediction is the in-depth understanding about the structural factors that affect algorithmic performance, which can also be considered as the guidance of the choice of algorithms when both the accuracy and complexity have to be taken into account. For example, if the network is highly clustered, the common-neighbor-based algorithms may be good choices since they can give relatively good prediction with very low complexity. And among all the known common-neighbor-based indices, the resource allocation index [51,52] seems the best. However, if the network is not highly clustered, or the distribution of the number of common neighbors decays too fast (like in the router-level Internet [56], 99.98% of node pairs share not more than two common neighbors), the common-neighbor-based algorithms are very poor, and we should try local path index [51,78] and local random walk index [79] that make use of more information. The optimal step for local random walk index is also determined by the structural feature: the longer the average distance, the larger the optimal step. And for a network with long average distance, directly applying the Kate index [66] may be the best choice. If the size of a network is not very large and this network has clear hierarchical or modular structure, the maximum likelihood method given organizing rules [80,18] may provide very accurate prediction.

Up until now, the studies of link prediction overwhelmingly emphasize on the unweighted undirected networks. For directed networks, even the ternary relations are complicated, and thus the simple common-neighbor-based similarity indices have to be modified to take into account the local motif structure [156]. Otherwise, even we can predict the existence of an arc between two nodes, we cannot determine its direction. In addition, the path-dependent similarity indices should also be extended to take into account the link direction [157]. The fundamental task of link prediction in weighted networks, namely to predict the existence of links with the help of not only the observed links but also their weights, has already been considered by Murata et al. [158] and Lü et al. [159]. The former [158] suggested that the links with higher weights are more important in predicting missing links, while the latter [159] indicated a completely opposite conclusion: the weak links play a more significant role. How to properly exploit the information of weights to improve the prediction accuracy is still an unsolved problem. A harder problem is to predict the weights of links, which is relevant to the traffic prediction for urban transportation and air transportation systems [160]. We are expecting that some variants of link prediction algorithms can also contribute to this domain.

A big challenge is the link prediction in multi-dimensional networks, where links could have different meanings. For example, a social network may consist of positive and negative links, respectively pointing to friends and foes [161], or trusted and distrusted peers [162]. Leskovec et al. [163] proposed a method to predict the signs of links (positive or negative), yet the prediction of both the existence of a link and its sign has not been well studied. Recent development of *social balance theory* may provide useful hints [164–166].

A more complicated kind of multi-dimensional network is the one consisted of several classes of nodes. For example, an online resource-sharing system, such as Del.icio.us,[20] can be represented by a network that consists of three kinds of nodes: users, URLs and tags. Different from the tripartite networks, nodes in the same class can also be connected, like an arc can be added from a user to her/his follower who has imposed her/his collections. Ignoring the connections within a class of nodes, the prediction of links between users and objects has already been investigated [167,168]. However, there is still nothing reported about the link prediction algorithms taking into account both the links within a class and the links between classes.

Inspired by the success in recommender systems, we think the prediction accuracy can be considerably improved by hybrid algorithms [169]. Given a specific target network, we can implement many individual prediction algorithms, and

[20] Del.icio.us is the largest social bookmarking system where a user is allowed to collect URLs as well as visit and impose other users' collections.

then try to select and organize them in a proper way. This so-called *ensemble learning* method can obtain better prediction performance than could be obtained from any of the individual algorithms [170]. Although the scientific significance of such a method is not clear to us, building ensemble systems for link prediction could be of huge practical value.

The algorithms' performance can be effectively enhanced by considering some external information, like the attributes of nodes [35]. In common sense, two people share more tastes and interests (and thus may of higher probability to be connected in a social network) if they have more common features, such as age, sex, job, and so on. The attribute information can be used to predict links without considering the network structures. Thus, when the existed links themselves are unreliable, attribute-based methods are preferable, which can to some extent solve the so-called cold start problem—a big challenge of link prediction [171]. Besides, community structures can also help to improve prediction accuracy [172]. In social networks, since one person may play different roles in different communities, the prediction in one domain can be inspired by the information in others [173]. For example, when we predict the collaborations between authors, we can consider their affiliations to improve the accuracy.

Most of current approaches take into account a single snapshot of a network to predict the missing or future links. Extensive experiments show that these methods well uncover whether a link exists. However, this static graph representation is difficult in predicting the repeated link occurrences. For example, it is impossible to predict whether and when two authors will collaborate again in co-authorship network. Addressing this problem, Huang and Lin [174] proposed a time-series link prediction approach considering the temporal evolutions of link occurrences, which is more appropriate for dealing with the link prediction problem in evolving networks, such as online social networks. Another way to involve time information is inspired by the fact that older events are less likely to be relevant to future links than recent ones. For example, author's interests may change over time and thus old publications might be less relevant to his current research area. Tylenda et al. [175] developed a graph-based link prediction method that incorporate the temporal information contained in evolving networks. They found that the performance can be improved by either time-based weighting of edges (i.e., giving the older events smaller weights or even neglecting them) or weighting of edges according to the connecting strength. However, to design effective algorithms and eventually settle this problem, we need in-depth and comprehensive understanding of temporal effects on human interests, attentions and so on, which asks for extensive empirical analyses.

Acknowledgements

We acknowledge Ci-Hang Jin, Hong-Kun Liu, Wei-Ping Liu, Ming-Sheng Shang, Qian-Ming Zhang and Yi-Cheng Zhang for their contributions to the collaborated works on link prediction, as well as Renaud Lambiotte, Medo Matuš, Roger Guimerà, Marco Saerens, Marta Sales-Pardo, Chi Ho Yeung and Zi-Ke Zhang for their valuable discussions, comments and suggestions. This work is partially supported by the National Natural Science Foundation of China under Grant Nos. 11075031 and 10635040, the Swiss National Science Foundation under Grant No. 200020-121848, and Shanghai leading discipline project under Grant No. S30501.

References

[1] R. Albert, A.-L. Barabási, Statistical mechanics of complex networks, Rev. Modern Phys. 74 (2002) 47.
[2] S.N. Dorogovtsev, J.F.F. Mendes, Evolution of networks, Adv. Phys. 51 (2002) 1079.
[3] M.E.J. Newman, The structure and function of complex networks, SIAM Rev. 45 (2003) 167.
[4] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.-U. Huang, Complex networks: structure and dynamics, Phys. Rep. 424 (2006) 175.
[5] L. da, F. Costa, F.A. Rodrigues, G. Travieso, P.R.U. Boas, Characterization of complex networks: a survey of measurements, Adv. Phys. 56 (2007) 167.
[6] G. Salton, M.J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, Auckland, 1983.
[7] G. Salton, Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley, Boston, 1989.
[8] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, New York, 2008.
[9] L. Getoor, C.P. Diehl, Link mining: a survey, ACM SIGKDD Explor. Newsl. 7 (2005) 3.
[10] H. Yu, et al., High-quality binary protein interaction map of the yeast interactome network, Science 322 (2008) 104.
[11] M.P.H. Stumpf, T. Thorne, E. de Silva, R. Stewart, H.J. An, M. Lappe, C. Wiuf, Estimating the size of the human interactome, Proc. Natl. Acad. Sci. USA 105 (2008) 6959.
[12] L.A.N. Amaral, A truer measure of our ignorance, Proc. Natl. Acad. Sci. USA 105 (2008) 6795.
[13] L. Schafer, J.W. Graham, Missing data: our view of the state of the art, Psychol. Methods 7 (2002) 147.
[14] G. Kossinets, Effects of missing data in social networks, Soc. Networks 28 (2006) 247.
[15] J.W. Neal, "Kracking" the missing data problem: applying Krackhardt's cognitive social structures to school-based social networks, Soc. Educ. 81 (2008) 140.
[16] C. von Mering, R. Krause, B. Snel, M. Cornell, S.G. Oliver, S. Field, P. Bork, Comparative assessment of large-scale data sets of protein-protein interactions, Nature 417 (2002) 399.
[17] C.T. Butts, Network inference, error, and information (in)accuracy: a Bayesian approach, Soc. Networks 25 (2003) 103.
[18] R. Guimerà, M. Sales-Pardo, Missing and spurious interactions and the reconstruction of complex networks, Proc. Natl. Acad. Sci. USA 106 (2009) 22073.
[19] S. Zhou, R.J. Mondragón, Accurately modeling the internet topology, Phys. Rev. E 70 (2004) 066108.
[20] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, E. Shir, A model of Internet topology using *k*-shell decomposition, Proc. Natl. Acad. Sci. USA 104 (2007) 11150.
[21] M. Sales-Pardo, R. Guimerà, L.A.N. Amaral, Extracting the hierarchical organization of complex systems, Proc. Natl. Acad. Sci. USA 104 (2007) 15224.
[22] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA 99 (2002) 7821.
[23] J. Shawe-Taylor, N. Cristianini, Kernels Methods for Pattern Analysis, Cambridge University Press, Cambridge, UK, 2004.
[24] M.E.J. Newman, Analysis of weighted networks, Phys. Rev. E 70 (2004) 056131.
[25] L. Breiman, P. Spector, Submodel selection and evaluation in regression: the *x*-random case, Int. Stat. Rev. 60 (1992) 291.

[26] R. Kohavi, A study of cross-valisation and bootstrap for accuracy estimation and model selection, in: Proceedings of the International Joint Conference on Artificial Intelligence, Morgan Kaufmann Publisher, Quebec, Canada, 1995, pp. 1137–1143.
[27] Y.-X. Zhu, L. Lü, Q.-M. Zhang, T. Zhou, Uncovering missing links with cold ends (unpublished).
[28] F. Wilcoxon, Individual comparisons by ranking methods, Biom. Bull. 1 (1945) 80.
[29] H.B. Mann, D.R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Ann. Math. Stat. 18 (1947) 50.
[30] J.A. Hanely, B.J. McNeil, The meaning and use of the area under a receiver operating characteristic (ROC) curve, Radiology 143 (1982) 29.
[31] S. Geisser, Predictive Inference: An Introduction, Chapman and Hall, New York, 1993.
[32] J.L. Herlocker, J.A. Konstann, K. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst. 22 (2004) 5.
[33] X. Su, T.M. Khoshgoftaar, A survey of collaborative filtering techniques, Adv. Artif. Intell. (2009) 421425.
[34] Z. Huang, X. Li, H. Chen, Link prediction approach to collaborative filtering, in: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries, ACM Press, New York, 2005.
[35] D. Lin, An information-theoretic definition of similarity, in: Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufman Publishers, San Francisco, 1998.
[36] E.A. Leicht, P. Holme, M.E.J. Newman, Vertex similarity in networks, Phys. Rev. E 73 (2006) 026120.
[37] D. Sun, T. Zhou, J.-G. Liu, R.-R. Liu, C.-X. Jia, B.-H. Wang, Information filtering based on transferring similarity, Phys. Rev. E 80 (2009) 017101.
[38] D.R. White, K.P. Reitz, Graph and semigroup homomorphisms on networks of relations, Soc. Networks 5 (1983) 193.
[39] P. Holme, M. Huss, Role-similarity based functional prediction in networked systems: application to the yeast proteome, J. R. Soc. Interface 2 (2005) 327.
[40] M.E.J. Newman, Clustering and preferential attachment in growing networks, Phys. Rev. E 64 (2001) 025102.
[41] P. Jaccard, Étude comparative de la distribution florale dans une portion des Alpes et des Jura, Bull. Soc. Vaud. Sci. Nat. 37 (1901) 547.
[42] T. Sørensen, A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons, Biol. Skr. 5 (1948) 1.
[43] E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, A.-L. Barabási, Hierarchical organization of modularity in metabolic networks, Science 297 (2002) 1551.
[44] M. Molloy, B. Reed, A critical point for random graphs with a given degree sequence, Random Structures Algorithms 6 (1995) 161.
[45] A.-L. Barabási, R. Albert, Emergence of scaling in random networks, Science 286 (1999) 509.
[46] Y.-B. Xie, T. Zhou, B.-H. Wang, Scale-free networks without growth, Physica A 387 (2008) 1683.
[47] P. Holme, B.J. Kim, C.N. Yoon, S.K. Han, Attack vulnerability of complex networks, Phys. Rev. E 65 (2002) 056109.
[48] C.-Y. Yin, W.-X. Wang, G.-R. Chen, B.-H. Wang, Decoupling process for better synchronizability on scale-free networks, Phys. Rev. E 74 (2006) 047102.
[49] G.-Q. Zhang, D. Wang, G.-J. Li, Enhancing the transmission efficiency by edge deletion in scale-free networks, Phys. Rev. E 76 (2007) 017101.
[50] L.A. Adamic, E. Adar, Friends and neighbors on the web, Soc. Networks 25 (2003) 211.
[51] T. Zhou, L. Lü, Y.-C. Zhang, Predicting missing links via local information, Eur. Phys. J. B 71 (2009) 623.
[52] Q. Ou, Y.-D. Jin, T. Zhou, B.-H. Wang, B.-Q. Yin, Power-law strength-degree correlation from resource-allocation dynamics on weighted networks, Phys. Rev. E 75 (2007) 021102.
[53] M.E.J. Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74 (2006) 036104.
[54] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks, Nature 393 (1998) 440.
[55] R. Ackland, Mapping the US political blogosphere: are conservative bloggers more prominent, in: Presentation to BlogTalk Downunder, Sydney, 2005, Available at: http://incsub.org/blogtalk/images/robertackland.pdf.
[56] N. Spring, R. Mahajan, D. Wetherall, T. Anderson, IEEE/ACM Trans. Netw. 12 (2004) 2.
[57] V. Batageli, A. Mrvar, Pajek datasets. Available at: http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm.
[58] D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks, J. Am. Soc. Inf. Sci. Technol. 58 (2007) 1019.
[59] M.T. Gastner, M.E.J. Newman, The spatial structure of networks, Eur. Phys. J. B 49 (2006) 247.
[60] H.-K. Liu, T. Zhou, Empirical study of Chinese city airline network, Acta Phys. Sinica 56 (2007) 106.
[61] S. Zhou, R.J. Mondragón, The rich-club phenomenon in the Internet topology, IEEE Commun. Lett. 8 (2004) 180.
[62] V. Colizza, A. Flammini, M.A. Serrano, A. Vespignani, Detecting rich-club ordering in complex networks, Nat. Phys. 2 (2006) 110.
[63] Y. Pan, D.-H. Li, J.-G. Liu, J.-Z. Liang, Detecting community structure in complex networks via node similarity, Physica A 389 (2010) 2849.
[64] Y.-L. Wang, T. Zhou, J.-J. Shi, J. Wang, D.-R. He, Empirical analysis of dependence between stations in Chinese railway network, Physica A 388 (2009) 2949.
[65] T. Zhou, J. Ren, M. Medo, Y.-C. Zhang, Bipartite network projection and personal recommendation, Phys. Rev. E 76 (2007) 046115.
[66] L. Katz, A new status index derived from sociometric analysis, Psychmetrika 18 (1953) 39.
[67] V.D. Blondel, A. Gajardo, M. Heymans, P. Senellart, P.V. Dooren, A measure of similarity between graph vertices: applications to synonym extraction and web searching, SIAM Rev. 46 (2004) 647.
[68] E.H. Moore, On the reciprocal of the general algebraic matrix, Bull. Amer. Math. Soc. 26 (1920) 394.
[69] R. Penrose, A generalized inverse for matrices, Proc. Cambridge Philos. Soc. 51 (1955) 406.
[70] D.J. Klein, M. Randic, Resistance distance, J. Math. Chem. 12 (1993) 81.
[71] F. Fouss, A. Pirotte, J.-M. Renders, M. Saerens, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, IEEE Trans. Knowl. Data. Eng. 19 (2007) 355.
[72] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107.
[73] H. Tong, C. Faloutsos, J.-Y. Pan, Fast random walk with restart and its applications, in: Proceedings of the 6th International Conference on Data Mining, IEEE Press, Washington, DC, USA, 2006, pp. 613–622.
[74] M.-S. Shang, L. Lü, T. Zhou, Y.-C. Zhang, Relevance is more significant than correlation: information filtering on sparse data, Europhys. Lett. 88 (2009) 68008.
[75] 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, ACM Press, New York, 2002, pp. 271–279.
[76] P. Chebotarev, E.V. Shamis, The matrix-forest theorem and measuring relations in small social groups, Autom. Remote Control 58 (1997) 1505.
[77] F. Fouss, L. Yen, A. Pirotte, M. Saerens, An experimental investigation of graph kernels on a collaborative recommendation task, in: Proceedings of the 6th International Conference on Data Mining, IEEE Press, Washington, DC, USA, 2006, pp. 863–868.
[78] L. Lü, C.-H. Jin, T. Zhou, Similarity index based on local paths for link prediction of complex networks, Phys. Rev. E 80 (2009) 046122.
[79] W. Liu, L. Lü, Link prediction based on local random walk, Europhys. Lett. 89 (2010) 58007.
[80] A. Clauset, C. Moore, M.E.J. Newman, Hierarchical structure and the prediction of missing links in networks, Nature 453 (2008) 98.
[81] A. Mantrach, N. van Zeebroeck, P. Francq, M. Shimbo, H. Bersini, M. Saerens, Semi-supervised classification and betweenness computation on large, sparse, directed, networks (unpublished).
[82] C. Zhou, L. Zemanová, G. Zamora, C.C. Hilgetag, J. Kurths, Hierarchical organization unveiled by functional connectivity in complex brain networks, Phys. Rev. Lett. 97 (2006) 238103.
[83] S. Redner, Teasing out the missing links, Nature 453 (2008) 47.
[84] G. Casella, R.L. Berger, Statistical Inference, Duxbury, Belmont, 2001.
[85] M.E.J. Newman, G.T. Barkema, Monte Carlo Methods in Statistical Physics, Clarendon, Oxford, 1999.
[86] V. Krebs, Mapping networks of terrorist cells, Connections 24 (2002) 43.
[87] H.A. Dawah, B.A. Hawkins, M.F. Claridge, Structure of the parasitoid communities of grass-feeding chalcid wasps, J. Anim. Ecol. 64 (1995) 708.

[88] M. Huss, P. Holme, Currency and commodity metabolites: their identification and relation to the modularity of metabolic networks, IET Syst. Biol. 1 (2007) 280.
[89] E. Mossel, E. Vigoda, Phylogenetic MCMC are misleading on mixtures of trees, Science 309 (2005) 2207.
[90] H.C. White, S.A. Boorman, R.L. Breiger, Social structure from multiple networks I: blockmodels of roles and positions, Am. J. Sociol. 81 (1976) 730.
[91] P.W. Holland, K.B. Laskey, S. Leinhardt, Stochastic blockmodels: first steps, Soc. Networks 5 (1983) 109.
[92] P. Dorelan, V. Batagelj, A. Ferligoj, Generalized Blockmodeling, Cambridge University Press, Cambridge, UK, 2005.
[93] E.M. Airoldi, D.M. Blei, S.E. Fienberg, X.P. Xing, Mixed-membership stochastic blockmodels, J. Mach. Learn. Res. 9 (2008) 1981.
[94] R. Guimerà, M. Sales-Pardo, L.A.N. Amaral, Classes of complex networks defined by role-to-role connectivity profiles, Nat. Phys. 3 (2007) 63.
[95] J. Reichardt, D.R. White, Role models for complex networks, Eur. Phys. J. B 60 (2007) 217.
[96] M.E.J. Newman, Assortative mixing in networks, Phys. Rev. Lett. 89 (2002) 208701.
[97] M.E.J. Newman, Mixing patterns in networks, Phys. Rev. E 67 (2003) 026126.
[98] R. Pastor-Satorras, A. Vázquez, A. Vespignani, Dynamical and correlation properties of the Internet, Phys. Rev. Lett. 87 (2001) 258701.
[99] A. Vázquez, R. Pastor-Satorras, A. Vespignani, Large-scale topological and dynamical properties of the Internet, Phys. Rev. E 65 (2002) 066130.
[100] T. Bayes, An essay towards solving a problem in the doctrine of chances, Philos. Trans. R. Soc. Lond. 53 (1763) 370.
[101] M. Metropolis, A.W. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculation by fast computing machines, J. Chem. Phys. 21 (1953) 1087.
[102] W. Zachary, An information flow model for conflict and fission in small groups, J. Anthropol. Res. 33 (1977) 452.
[103] D. Lusseau, et al., The bottlenose dolphin community of Doubtful sound features a large proportion of long-lasting associations, Behav. Ecol. Sociobiol. 54 (2003) 396.
[104] R. Guimerà, S. Mossa, A. Turtschi, L.A.N. Amaral, The worldwide air transportation network: anomalous centrality, community structure, and cities' global roles, Proc. Natl. Acad. Sci. USA 102 (2005) 7794.
[105] J.G. White, E. Southgate, J.N. Thomson, S. Brenner, The structure of the nervous system of the nematode C. elegans, Philos. Trans. R. Soc. Lond. Ser. B 314 (1986) 1.
[106] J.L. Reed, T.D. Vo, C.H. Schilling, B.Ø Palsson, An expanded genome-scale model of Escherichia coli K-12 (iJR904 GSM/GPR), Genome Biol. 4 (2003) R54.
[107] N. Friedman, L. Getoor, D. Koller, A. Pfeffer, Learning probabilistic relational models, in: Proceedings of the 16th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 1999, p. 1300.
[108] D. Heckerman, C. Meek, D. Koller, Probabilistic entity-relationship models, PRMS, and plate models, in: Proceedings of the 21st International Conference on Machine Learning, Banff, Canada, 2004, p. 55.
[109] K. Yu, W. Chu, S. Yu, V. Tresp, Z. Xu, Stochastic relational models for discriminative link prediction, in: Proceedings of Neural Information Precessing Systems, MIT Press, Cambridge, MA, 2007, pp. 1553–1560.
[110] J. Neville, Statistical models and analysis techniques for learning in relational data, Ph.D. Thesis, 2006.
[111] D. Heckerman, C. Meek, D. Koller, Probabilistic models for relational data, Tech. Rep. MSR-TR-2004-30, Microsoft Research, 2004.
[112] D. Heckerman, D. Geiger, D. Chickering, Learning Bayesian networks: the combination of knowledge and statistical data, Mach. Learn. 20 (1995) 197.
[113] B. Taskar, P. Abbeel, D. Koller, Discriminative probabilistic models in relational data, in: Preceedings of the 18th Conference on Uncertainty in Artificial Intelligence, UAI02, Edmonton, Canada, 2002, p. 485.
[114] B. Taskar, M.-F. Wong, P. Abbeel, D. Koller, Link prediction in relational data, in: Proceedings of Neural Information Precessing Systems, MIT Press, Cambridge, MA, 2004, p. 659.
[115] D. Heckerman, D. Chickering, C. Meek, R. Rounthwaite, C. Kadie, Dependency networks for inference, collaborative filtering, and data visualization, J. Mach. Learn. Res. 1 (2000) 49.
[116] J. Neville, D. Jensen, Relational dependency networks, J. Mach. Learn. Res. 8 (2007) 653.
[117] G. Casella, E.I. George, Explaining the Gibbs sampler, Amer. Statist. 46 (3) (1992) 167.
[118] Z. Xu, V. Tresp, K. Yu, S. Yu, H.-P. Kriegel, Dirichlet enhanced relational learning, in: Proceedings of the 22nd Internatonal Conference on Machine Learning, Bonn, Germany, 2005, p. 1004.
[119] W. Buntine, Operations for learning with graphical models, J. Artificial Intelligence Res. 2 (1994) 159.
[120] D. Spiegelhalter, Bayesian graphical modeling: a case-study in monitoring health outcomes, Appl. Stat. 47 (1998) 115.
[121] K. Yu, W. Chu, Gaussian process models for link analysis and transfer learning, in: Proceedings of Neural Information Precessing Systems, MIT Press, Cambridge, MA, 2007, p. 1657.
[122] W. Chu, V. Sindhwani, Z. Ghahramani, S.S. Keerthi, Relational learning with Gaussian processes, in: Proceedings of Neural Information Precessing Systems, MIT Press, Cambridge, MA, 2006, p. 289.
[123] J. O'Madadhain, J. Hutchins, P. Smyth, Prediction and ranking algorithms for event-based network data, in: Proceedings of SIGKDD 2005, ACM Press, New York, 2005, p. 23.
[124] M.-S. Shang, L. Lü, Y.-C. Zhang, T. Zhou, Empirical analysis of web-based user-object bipartite networks, Europhys. Lett. 90 (2010) 48006.
[125] J. Kunegis, E.W. De Luca, S. Albayrak, The link predection problem in bipartite networks. arXiv:1006.5367.
[126] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J.R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proc. Natl. Acad. Sci. USA 107 (2010) 4511.
[127] W. Zeng, M.-S. Shang, Q.-M. Zhang, L. Lü, T. Zhou, Can dissimilar users contribute to accuracy and diversity of personalized recommendation, Internat. J. Modern Phys. C 21 (2010) 1217.
[128] Q.-M. Zhang, M.-S. Shang, W. Zeng, Y. Chen, L. Lü, Empirical comparison of local structural similarity indices for collaborative-filtering-based recommender systems, Physics Procedia 3 (2010) 1887.
[129] J. Schafer, J. Konstan, J. Riedl, E-commerce recommendation applications, Data Min. Knowl. Discov. 5 (2001) 115.
[130] Z. Huang, D.D. Zeng, A link prediction approach to anomalous email detection, in: Proceedings of 2006 IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, 2006, p. 1131.
[131] B. Gallagher, H. Tong, T. Eliassi-Rad, C. Faloutsos, Using ghost edges for classification in sparsely labeled networks, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, 2008, p. 256.
[132] K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A.A. Nanavati, A. Joshi, Social ties and their relevance to churn in mobile telecom networks, in: Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology, ACM Press, New York, 2008, p. 668.
[133] M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2004) 026113.
[134] R. Guimerà, A. Díaz-Guilera, F. Vega-Redondo, A. Cabrales, A. Arenas, Optimal network topologies for local search with congestion, Phys. Rev. Lett. 89 (2002) 248701.
[135] G. Yan, T. Zhou, B. Hu, Z.-Q. Fu, B.-H. Wang, Efficient routing on complex networks, Phys. Rev. E 73 (2006) 046108.
[136] M. Barahona, L.M. Pecora, Synchronization in small-world systems, Phys. Rev. Lett. 89 (2002) 054101.
[137] A. Arenas, A. Díza-Guilera, J. Kurths, Y. Moreno, C. Zhou, Phys. Rep. 469 (2008) 93.
[138] R. Pastor-Satorras, A. Vespignani, Epidemics and immunization in scale-free networks, in: S. Bornholdt, H.G. Schuster (Eds.), Handbook of Graphs and Networks, Wiley-VCH, Berlin, 2003.
[139] T. Zhou, Z.-Q. Fu, B.-H. Wang, Epidemic dynamics on complex networks, Prog. Nat. Sci. 16 (2006) 452.
[140] G. Caldarelli, A. Capocci, P. De Los Rios, M.A. Muñoz, Scale-free networks from varying vertex intrinsic fitness, Phys. Rev. Lett. 89 (2002) 258702.
[141] S. Valverde, R.F. Cancho, R.V. Solé, Scale-free networks from optimal design, Europhys. Lett. 60 (2002) 512.
[142] M. Baiesi, S.S. Manna, Scale-free networks from a Hamiltonian dynamics, Phys. Rev. E 68 (2003) 047103.
[143] B.J. Kim, A. Trusina, P. Minnhagen, K. Sneppen, Self organized scale-free networks from merging and regeneration, Eur. Phys. J. B 43 (2005) 369.

[144] J.I. Perotti, O.V. Billoni, F.A. Tamarit, D.R. Chialvo, S.A. Cannas, Emergent self-organized complex network topology out of stability constraints, Phys. Rev. Lett. 103 (2009) 108701.
[145] G. Bianconi, P. Pin, M. Marsili, Assessing the relevance of node features for network structure, Proc. Natl. Acad. Sci. USA 106 (2009) 11433.
[146] H.-K. Liu, T. Zhou, Review on the studies of airline networks, Prog. Nat. Sci. 18 (2008) 601.
[147] A.-X. Cui, Y. Fu, M.-S. Shang, D.-B. Chen, T. Zhou, Emergence of local structures in complex network: common neighborhood drives the network evolution, Acta Phys. Sinica 60 (2011) 30.
[148] W.-K. Xiao, J. Ren, F. Qi, Z.-W. Song, M.-X. Zhu, H.-F. Yang, H.-Y. Jin, B.-H. Wang, T. Zhou, Emprical study on clique-degree distribution of networks, Phys. Rev. E 76 (2007) 037102.
[149] R. Lambiotte, V.D. Blondel, C. de Kerchove, E. Huens, C. Prieur, Z. Smoreda, P. Van Dooren, Geographical dispersal of mobile communication networks, Physica A 387 (2008) 5317.
[150] W.-S. Jung, F. Wang, H.E. Stanley, Gravity model in the Korean highway, Europhys. Lett. 81 (2008) 48005.
[151] P. Kaluza, A. Koelzsch, M.T. Gastner, B. Blasius, The complex network of global cargo ship movements, J. R. Soc. Interface 7 (2010) 1093.
[152] H.-K. Liu, X.-L. Zhang, L. Cao, B.-H. Wang, T. Zhou, Analysis on the connecting mechanism of Chinese city airline network, Sci. China Ser. G 39 (2009) 935.
[153] C.W.J. Granger, Investigating causal relations by econometric models and cross-spectral methods, Econometrica 37 (1969) 424.
[154] H.-K. Liu, X.-L. Zhang, T. Zhou, Structure and external factors of Chinese city airline network, Physics Procedia 3 (2010) 1781.
[155] Q.-M. Zhang, M.-S. Shang, L. Lü, Similarity-based classification in partially labeled networks, Internat. J. Modern Phys. C 21 (2010) 813.
[156] U. Alon, Network motifs: theory and experimental approaches, Nat. Rev. Genet. 8 (2007) 450.
[157] A. Mantrach, L. Yen, J. Callut, K. Françoisse, M. Shimbo, M. Saerens, The sum-over-paths covariance kernel: a novel covariance measure between nodes of a directed graph, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 1112.
[158] T. Murata, S. Moriyasu, Link prediction of social networks based on weighted proximity measure, in: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, ACM Press, New York, 2007.
[159] L. Lü, T. Zhou, Link prediction in weighted networks: the role of weak ties, Europhys. Lett. 89 (2010) 18001.
[160] H. Yin, S.C. Wong, J. Xu, C.K. Wong, Urban traffic flow prediction using a fuzzy-neural approach, Transp. Res. C 10 (2002) 85.
[161] J. Kunegis, A. Lommatzsch, C. Bauckhage, The slashdot zoo: mining a social network with negative edges, in: Proceedings of WWW'2009, ACM Press, New York, 2009.
[162] R.V. Guha, R. Kumar, P. Raghavan, A. Tomkins, Propagation of trust and distrust, in: Proceedings of WWW'2004, ACM Press, New York, 2004.
[163] J. Leskovec, D. Huttenlocher, J. Kleinberg, Predicting positive and negative links in online social networks, in: Proceedings of WWW'2010, ACM Press, New York, 2010.
[164] V.A. Traag, J. Bruggeman, Community detection in networks with positive and negative links, Phys. Rev. E 80 (2009) 036115.
[165] S.A. Marvel, S.H. Strogatz, J.M. Kleinberg, Energy landscape of social balance, Phys. Rev. Lett. 103 (2009) 198701.
[166] M. Szell, R. Lambiotte, S. Thurner, Multirelational organization of large-scale social networks in an online world, Proc. Natl. Acad. Sci. USA 107 (2010) 13636.
[167] Z.-K. Zhang, T. Zhou, Y.-C. Zhang, Personalized recommendation via integrated diffusion on user–item–tag tripartite graphs, Physica A 389 (2010) 179.
[168] 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) 28002.
[169] R. Burke, Hybrid recommender systems: survey and experiments, User Model. User-Adapt. Interact. 12 (2002) 331.
[170] R. Polikar, Ensemble based systems in decision making, IEEE Circuits Syst. Mag. 6 (3) (2006) 21.
[171] V. Leroy, B.B. Cambazoglu, F. Bonchi, Cold start link prediction, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, 2010, p. 393.
[172] E. Zheleva, L. Getoor, J. Golbeck, Ugur Kuter, Using friendship ties and family circles for link prediction, in: Proceedings of the 2nd Workshop on Social Network Mining and Analysis, ACM Press, New York, 2008.
[173] B. Cao, N.N. Liu, Q. Yang, Transfer learning for collective link prediction in multiple heterogenous domains, in: Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 2010.
[174] Z. Huang, D.K.J. Lin, The time-series link prediction problem with applications in communication surveillance, INFORMS J. Comput. 21 (2009) 286.
[175] T. Tylenda, R. Angelova, S. Bedathur, Towards time-aware link prediction in evolving social networks, in: Proceedings of the 3rd Workshop on Social Network Mining and Analysis, ACM Press, New York, 2009.