Link Weight Prediction Using Supervised Learning Methods and Its Application to Yelp Layered Network

Chenbo Fu, Minghao Zhao, Lu Fan, Xinyi Chen, Jinyin Chen, Zhefu Wu, Yongxiang Xia, *Senior Member, IEEE*, and Qi Xuan

Abstract—Real-world networks feature weights of interactions, where link weights often represent some physical attributes. In many situations, to recover the missing data or predict the network evolution, we need to predict link weights in a network. In this paper, we first proposed a series of new centrality indices for links in line graph. Then, utilizing these line graph indices, as well as a number of original graph indices, we designed three supervised learning methods to realize link weight prediction both in the networks of single layer and multiple layers, which perform much better than several recently proposed baseline methods. We found that the resource allocation index (RA) plays a more important role in the weight prediction than other topological properties, and the line graph indices are at least as important as the original graph indices in link weight prediction. In particular, the success application of our methods on Yelp layered network suggests that we can indeed predict the offline co-foraging behaviors of users just based on their online social interactions, which may open a new direction for link weight prediction algorithms, and meanwhile provide insights to design better restaurant recommendation systems.

Index Terms—Complex network, link weight prediction, structural feature, layered network, machine learning

1 Introduction

Many complex systems in sociology, biology, and computer science can be represented by networks, where the nodes and links capture the structure of these real-world systems in various ways [1], [2], [3], [4], [5], [6], [7]. In the past decades, stimulated by the collection of massive structural data and the discovery of abundant phenomena for many networked systems, a surge of studies have been performed to study network structure, and thus *network science*, as a new frontier interdiscipline, emerges.

In network science, a series of structural properties around nodes and links have been proposed, including node centrality [8], clustering coefficient [9], assortativity [10], similarity between pairwise nodes [11], and so on. These properties are the basis of many network models, such as small-world [9], scale-free [12], modular and hierarchy [13] networks. Besides, they capture certain local topological information of systems, and thus can be used to design network algorithms. Typically, node centrality is always used to measure the individual importance in a system. For example, Xuan et al. [14] utilized node degree in a temporal email network to predict the first technical contribution of a developer in Open Source Software (OSS) projects. They found that such naive algorithm behaves even better than PageRank [15] and Hits [16] algorithms. Liben-Nowell and Kleinberg [17] adopted a number of similarity metrics between pairwise nodes in a social network to predict new interactions between them. By comparing with the random prediction, they found that information about future interactions can indeed be extracted from network topology alone. Such node similarity can also be used to detect community structure in networks [18]. More recently, Xuan and Wu [19] defined node similarity between layered networks and used it to design node matching algorithms. However, the original node matching algorithm has relatively high time complexity. Xuan et al. [20] then further proposed an iterative algorithm to increase the efficiency. They found that nodes of higher degree play more important roles, especially in scale-free networks, i.e., better matching results can be obtained, given the nodes of higher degree as the revealed matched nodes beforehand.

Real-world networks, e.g., social networks, are always highly dynamic, i.e., the temporal network structure grows and changes quickly overtime through the addition of new links [21], [22]. Understanding the mechanisms by which they evolve is a fundamental question that motivates the design of network models, and also link prediction algorithms. Moreover, in biology, it is always expensive and labor-intensive to detect all the interactions between huge number of genes by experiments, and thus link prediction may be adopted to complement missing links and thus

- *C. Fu, M. Zhao, L. Fan, X. Chen, J. Chen, Z. Wu, and Q. Xuan are with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China. E-mail: {cbfu, chenjinyin, wzf, xuanqi}@zjut.edu.cn, Yzbyzmh1314@163.com, float.hellowmr@gmail.com, adebut@126.com.*
- *Y. Xia is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China. E-mail: xiayx@zju.edu.cn.*

Manuscript received 4 July 2017; revised 18 Jan. 2018; accepted 29 Jan. 2018. Date of publication 5 Feb. 2018; date of current version 5 July 2018.
(Corresponding author: Qi Xuan.)
Recommended for acceptance by L.B. Holder.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2018.2801854

decrease the cost. For example, Chen and Liu [23] predicted the protein-protein interaction by random decision forest framework, and Yang et al. [24] applied link prediction algorithm to predict the missing links of gene association meta-networks. Generally, link prediction is based on network structure, which attempts to uncover the missing links or predict the future interactions between pairwise nodes from the current network structure. Link prediction is also served as a significant technique in recommender systems, e.g., friendship or online shopping recommendation [25], [26], [27], [28], [29]. Besides the study of Liben-Nowell and Kleinberg [17], there were a bunch of work studying link prediction in the past decades. Zhou et al. [30] compared a number of local similarity indices on several disparate networks, and found that the algorithm behaves best when using *Resource Allocation* (RA) index. Hassan et al. [31] adopted a number of supervised learning methods, taking a set of features as input. They found that *Support Vector Machine* (SVM) performs best and there are always a small subset of features playing a significant role in link prediction. Lichtenwalter et al. [32] investigated the issues such as network observational period, variance reduction, topological causes and degree of imbalance, and sampling approaches, motivating the use of a supervised framework. Based on this, they presented an effective flow-based predicting algorithm which outperforms unsupervised methods by more than 30 percent AUC. Scellato et al. [33] studied the link prediction on location-based social networks, and found that the inclusion of information about places and related user activities can increase the algorithm performance. More studies on link prediction can be found in a recent survey [11].

Many previous studies on link prediction just focused on unweighted networks, while few of them tried to utilize or just estimate the weights of links. In fact, many real-world networks are weighted networks, where each link (i, j) has a unique weight w_{ij} associating with link attributes [34], [35], [36]. For example, in brain networks, link weight represents the strength of connection [3]; in protein-protein interaction networks, link weight stands for the interaction confidence score [7]; in airline networks, link weight denotes the number of flights [1], [37]; and in social networks, link weight captures the strength of friendship [38]. Murata et al. [39] proposed an improved method to predict links based on weighted proximity measures. Their method is based on an assumption that proximities between nodes can be estimated better by using both graph proximity measures and the weights of existing links in a social network. Lü and Zhou [40] used local weighted similarity indices to estimate the likelihood of the existence of links in weighted networks. They found that the weak ties sometimes play a significant role in the link prediction. Recently, Backstrom and Kleinberg [41] tried to identify strong social links, i.e., spouses or romantic partners, within an individual's network neighborhood, which can be considered as a link weight prediction problem. They developed a new measure of link strength, namely *dispersion*, capturing the extent to which two people's mutual friends are not themselves well-connected, and found this new measure is a relatively strong indicator of romantic relationship. Another typical scene is that, nowadays, many online systems provide social networks to strengthen the interactions between customers. While there is an explicit social structure, we want to know whether such social activities can lead the involved individuals make technical or commercial contributions on similar items, e.g., committing to the same files when developing software in OSS projects, visiting and reviewing the same restaurants on Yelp, and so on. Such social and technical congruence [42], [43] has been revealed in some previous studies and has the potential to design better recommender system in both social and technical sides. In this paper, we would like to first project the bipartite technical contribution network on people side and establish the weighted collaboration network, and then utilize the social network structure to design link weight prediction algorithm to predict the link weight in collaboration network.

Link weight prediction is a relatively new topic. Recently, Aicher et al. [44] developed a weighted stochastic block model, which can be applied to infer both the existence and weights of links. Zhao et al. [45] proposed a method based on reliable routes to extend unweighted similarity indices to weighted ones, which can be used to predict the weights of links by assuming that similarity scores are linearly correlated with link weights. Zhu et al. [46] developed a novel method to predict link weights by examining the network structure surrounding a node, with the assumption that the formation of link weight is regulated by local clusterings in which homogeneous links tend to have similar weights. However, these works are just based on a single proximity metric but discard many other useful information. Hially et al. [47], on the other hand, integrated link weight information into their supervised learning methods for link prediction. They found that incorporating the weight information makes the methods have better performance, but only with a slight difference.

Here, we treat the link weight prediction as a supervised regression problem, which thus is different from the romantic relationship identifying problem [41] and the link prediction utilizing supervised learning by integrating link weight information [47], since both of them can be considered as classification problems. Therefore, link weight prediction can also be solved in the framework of supervised learning. The main contributions of the paper are as follows:

- First, we transform an original unweighted network to a line graph [48]. The nodes in the line graph represent the links in the original graph, and two nodes are connected in the line graph if the corresponding links share the same terminal node in the original graph. We then utilize the node centrality indices in the line graph to define the importance of links in the original graph.
- Second, we extract two groups of features including original graph features and line graph features. The original graph features contain most similarity features which can be viewed as the features of pairwise nodes associated with links. The line graph features contain the centrality features which can be viewed as directly edge features. Then we utilize them to establish supervised learning algorithms, and the results show such algorithms outperform the baseline methods. The experiments show that original graph features and line graph features complement each other. Furthermore, we also investigate the time complexity of feature extraction.

- Third, we established a Yelp layered network, capturing both online friendships and offline foraging behaviors, where the links denote friendships and weights stand for the times that two customers have visited the same restaurants. We further use the topological information obtained from online social links to estimate the link weights. This dataset can be used as a benchmark to test link weight prediction across layered networks.

The rest of paper is organized as follows. In Section 2, we make a brief description of the graph model and two performance metrics. In Section 3, we introduce all feature indices that will be utilized to design supervised learning algorithms. In particular, we introduced how to transfer an original unweighted network to a line graph, and utilize the node centrality indices in the line graph to define the importance of links in the original graph. Link weight prediction experiments on several benchmark real-world networks are shown in Section 4. We apply our supervised learning methods on Yelp layered network in Section 5, and validate that such link weight prediction methods perform well even across layered networks. Finally, the paper is concluded and discussed in Section 6.

2 Graph Model and Performance Metrics

An undirected and weighted network is modeled by a graph $G(V, E, W)$, where V, E and W are sets of nodes, links and weights, respectively. For each link $(i, j) \in E$, the weight is denoted by $w_{i,j}$, with $w_{i,j} = w_{j,i}$, since we didn't consider the direction of link. We randomly divide the weight set W into two parts: the training set W_T and the test set W_V, where $W_T \cup W_V = W$ and $W_T \cap W_V = \emptyset$. In this paper, we use the two traditional metrics in this area to measure the goodness of fit, i.e., Pearson Correlation Coefficient (PCC) and Root Mean Squared Error (RMSE), defined as following:

- *Pearson Correlation Coefficient*. The definition of PCC is

$$PCC = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{x_i - \overline{x}}{s_x}\right)\left(\frac{y_i - \overline{y}}{s_y}\right), \tag{1}$$

  where n is the sample size, $\overline{x}$ and s_x are the mean and the standard deviation of n samples of variable x, and $\overline{y}$ and s_y are the mean and the standard deviation of n samples of variable y, respectively. PCC is a measure of the linear correlation between two variables x and y. We have $PCC \in [-1, 1]$. Two variables x and y are considered positively correlated if $PCC > 0$, negatively correlated if $PCC < 0$, and not correlated if $PCC = 0$.
- *Root Mean Squared Error*. The defination of RMSE is

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{n}}, \tag{2}$$

  where y_i is the real response in the test set and $\widehat{y}_i$ is the corresponding estimation given by the learning model.

3 Feature Indices

In this work, we extract two groups of features: original graph features and line graph features. In original graph, similarity indices are often used for link prediction that attempts to estimate the likelihood of the existence of a link, and have been proposed in many empirical studies on social networks [11], [49]. The likelihood is often associated with the similarity of pairwise nodes. However, in many real-world networks, the weights of links may have their own physical meanings which might not be captured by the similarity between the associated nodes. Therefore, in this study, we first transform original graphs to line graphs, and then extract the edge features in original graphs directly by using the centrality indices in line graphs.

3.1 Original Graph

In original graph, similarity indices are directly defined as how many common features two nodes share [50]. Considering a pair of nodes, namely i and j, we assign a score s_{ij} for the similarity index between them [11]. In addition, we also calculate edge betweenness [51] as a supplement. In particular, the features in original graph include:

- Common Neighbors (CN). It is defined as

$$s_{ij}^{CN} = |\Gamma(i) \cap \Gamma(j)|, \tag{3}$$

  where $\Gamma(i)$ denotes the set of neighbors of node i and $|\bullet|$ is the cardinality of set. In network theory, it is easy to calculate CN index by the adjacency matrix A, i.e., $s_{ij}^{CN} = (A^2)_{ij}$, where the element in the matrix $a_{ij} = 1$ if node i and node j are connected, and $a_{ij} = 0$ otherwise. In a social network, it is reasonable that two individuals are likely to be friends if they share many common friends [52], [53]. Certainly, there are many other similar metrics based on common neighbors, but with the different normalization methods [11], as presented one by one in the following.
- Salton Index (SA). It is defined as

$$s_{ij}^{SA} = \frac{|\Gamma(i) \cap \Gamma(j)|}{\sqrt{k_i \times k_j}}, \tag{4}$$

  where k_i and k_j denote the degree of node i and node j, respectively. This similarity index is also known as the cosine similarity [54].
- Jaccard Index (JAC). It is defined as

$$s_{ij}^{JAC} = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|}. \tag{5}$$

  Jaccard is a classical statistical parameter used for comparing the similarity or diversity of sample sets [55].
- Hub Promoted Index (HPI). It is defined as

$$s_{ij}^{HPI} = \frac{|\Gamma(i) \cap \Gamma(j)|}{\min(k_i, k_j)}. \tag{6}$$

  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 [56].

- Hub Depressed Index (HDI). It is defined as

$$s_{ij}^{HDI} = \frac{|\Gamma(i) \cap \Gamma(j)|}{\max(k_i, k_j)}. \quad (7)$$

This index is analogously to the above index but has an opposite consideration [11].

- $S\phi$rensen Index (SI). It is defined as

$$s_{ij}^{SI} = \frac{2|\Gamma(i) \cap \Gamma(j)|}{k_i + k_j}. \quad (8)$$

This index is a compromise of the above two and consider the average degree of nodes i and j, which is often used for ecological community data [57].

- Leicht-Holme-Newman Index (LHN). It is defined as

$$s_{ij}^{LHN} = \frac{|\Gamma(i) \cap \Gamma(j)|}{k_i \times k_j}. \quad (9)$$

This index is similar to Salton Index, but assigns even smaller similarity to the pairwise nodes of larger degree [58].

- Adamic-Adar Index (AA). It is defined as

$$s_{ij}^{AA} = \sum_{z \in \Gamma(i) \cap \Gamma(j)} \frac{1}{\log k_z}, \quad (10)$$

the main assumption of this index is that the common neighbors of smaller degree contribute more to the similarity. For example, in a social network, many people may know a famous man, but they themselves may not know each other [59].

- Resource Allocation Index. It is defined as

$$s_{ij}^{RA} = \sum_{z \in \Gamma(i) \cap \Gamma(j)} \frac{1}{k_z}. \quad (11)$$

RA index is close to AA, but punish more to the common neighbors of higher degree. In some cases, it was shown that RA performs better than AA in link prediction [30], [60].

- Preferential Attachment Index (PA). It is defined as

$$s_{ij}^{PA} = k_i \times k_j. \quad (12)$$

The mechanism of preferential attachment can be used to generate scale-free networks, where the probability that a new link connect to the node i is proportional to the node's degree k_i [12]. It is shown that this index is a very significant feature in link prediction [31], [61].

- Friends-Measure (FM). It is defined as

$$s_{ij}^{FM} = \sum_{u \in \Gamma(i)} \sum_{v \in \Gamma(j)} \delta(u, v), \quad (13)$$

where $\delta(u, v)$ equals 1 when nodes u and v are the same node or there is a link between them, and equals 0 otherwise. FM expands CN a little bit, and increase the similarity by considering the links between the common neighbors [61].

- Local Path Index (LP). It is defined as

$$s_{ij}^{LP} = (A^2)_{ij} + \varepsilon (A^3)_{ij}, \quad (14)$$

where ε is a free parameter, in this paper, we set $\varepsilon = 0.1$. A is the adjacency matrix, and $(A^k)_{ij}$ is the number of path connecting the nodes i and j with length k [11]. LP considers the local path between pairwise nodes, and can get wider horizon than the indices just based on common neighbors [30].

- Local Random Walk (LRW). LRW considers the finite-step random walk on network [62]. Assume a random walker starts from node i, and $\rho_{ij}(t)$ is the probability that the walker arrives at node j at time step t, then we can get $\rho_{ij}(t+1) = \mathbf{P}^T \rho_{ij}(t)$, where $\mathbf{P}^T$ is the transition matrix with $P_{ij} = 1/k_i$, if node i and node j are connected and $P_{ij} = 0$ otherwise. The initial value $\rho_{ij}(0)$ can be represented by an $N \times 1$ vector with ith element equals to 1 and others equal to 0. Thus LRW at time step t is defined as

$$s_{ij}^{LRW}(t) = q_i \rho_{ij}(t) + q_j \rho_{ji}(t), \quad (15)$$

where q_i and q_j are the free parameters. In this work, we set $t = 10N$ and $q_i = k_i$ for $i = 1, 2, \ldots, N$.

- Edge Betweenness (EB). It is defined as

$$EB_{ij} = \sum_{s,t \in V; s \neq t} \frac{n_{st}^{(i,j)}}{g_{st}}, \quad (16)$$

where g_{st} is the number of shortest paths between nodes s and t, and $n_{st}^{(i,j)}$ is the number of shortest paths between nodes s and t that pass through the edge (i, j) [51].

In network science, node centrality is often used to identify the important nodes [63], e.g., finding the essential people in social network, or key spreaders in epidemics. In order to investigate the importance of links, here, we transform the original unweighted network $G = (V, E)$ to line graph $L(G) = (E, D)$ [48], then utilize the node centrality in line graph to define the importance of links in original graph. In this representation, the node in the line graph is the link in the original graph, and two nodes have a connection between them if the corresponding links share the same terminal node in the original graph. As shown in Fig. 1, we transform the links (i, j), (j, l) and (j, m) in the original graph to nodes a, b and c in the line graph. Nodes a and b are connected since the links (i, j) and (j, l) have the same terminal node j, so are the connections between a, b and c.

Based on this transformation, we define the line graph features by node centrality indices in line graph, and then use them to capture the importance of links in the original graph directly.

3.2 Line Graph

- Degree Centrality (DC). It is defined as

$$DC_i = \frac{k_i}{N - 1}, \quad (17)$$

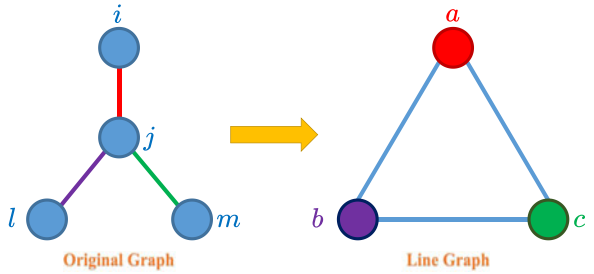


Fig. 1. A schematic diagram for the transformation from original graph to line graph, where the edges (i, j), (j, l) and (j, m) in original graph are transformed to nodes a, b and c in line graph, respectively.

where k_i is the degree of node i and N is the total number of nodes in the line graph [64].

- Closeness Centrality (CC). It is defined as

$$CC_i = \frac{N}{\sum_{j=1}^{N} d_{ij}}, \tag{18}$$

where d_{ij} denotes the shortest path length between nodes i and j in the line graph. The shorter the distances between node i and the rest nodes are, the more central the node i is, and thus the larger CC_i index is [65].

- Betweenness Centrality (BC). It is defined as

$$BC_i = \sum_{s \neq i \neq t} \frac{n_{st}^i}{g_{st}}, \tag{19}$$

where g_{st} is the total number of shortest paths between nodes s and t in the line graph, and n_{st}^i represents the number of shortest paths between nodes s and t that pass through node i [66].

- Eigenvector Centrality (EC). Eigenvector Centrality is also known as eigencentrality [67]. It is defined as

$$EC_i = \alpha \sum_{j=1}^{N} a_{ij} EC_j, \tag{20}$$

where a_{ij} is the element of the adjacency matrix of the line graph, i.e., $a_{ij} = 1$ if nodes i and j are connected and $a_{ij} = 0$ otherwise, and α should be less than the reciprocal of maximum eigenvalue of the adjacency matrix.

- PageRank (PR). PageRank is a popular way of measuring the importance of website pages [15]. The underlying assumption is that more important webpages tend to receive more links from other webpages. Its iterative formula is defined as

$$PR_i(t) = (1 - c) \sum_{j=1}^{N} a_{ji} \frac{PR_j(t-1)}{k_j} + \frac{c}{N}, \tag{21}$$

where c is a free parameter between zero and one. In this study, we set $c = 0.15$.

- Clustering Coefficient (C). In this case, we consider the local clustering coefficient. It can be defined as [9]

$$C_i = \frac{2L_i}{k_i(k_i - 1)}, \tag{22}$$

where L_i is the number of links between the k_i neighbors of node i.

- H-index (H). H-index is a popular metric which is used to measure both the productivity and citation impact of a scholar or scientist [68]. Recently, Lü et al. [69] extended this concept to networks, i.e., assuming node i to be a scholar and its neighbors to be the papers of node i, and a neighbor's degree to be the citations of each paper. Sorting the degree of node i's neighbor by decreasing order, the H-index can be calculated as following:

$$H_i = \max_{j \in \Gamma(i)} \min(k_j, j). \tag{23}$$

- Coreness (CO). The coreness is defined based on k-core. The k-core of a network is defined as the maximal subnetwork where every node in the subnetwork has at least degree k [70], [71], and if a node belongs to k-core but not $(k+1)$-core, then it has coreness k [72].

4 Experiments and Results

4.1 Data Description

In this study, we use the following six weighted networks as benchmarks to test link weight prediction methods.

- *Celegans* is a neural network of the nematode worm C. elegans, where nodes represent neurons and links stand for synaptic contacts, and the weight of a link is the number of synapses between two neurons [9].
- *USAir* is a network of US air transportation, where nodes and links represent airports and flights between pairwise airports, respectively. The weight of a link denotes the frequency of flights between the corresponding airports [73].
- *Lesmis* is a network of characters in Victor Hugo's famous novel *Les Miserables*. The nodes are the characters and two nodes are connected if the corresponding characters co-appear in the same chapter of the book. The weight of a link indicates the frequency of such co-appearance [74].
- *NetScience* is a network for scientist collaborations in the area of network science. The nodes and links represent scientists and co-author relationships, respectively. The weight of a link represents the number of papers that the corresponding two scientists co-authored [1], [2].
- *Geom* is also a collaboration network, but in the area of computational geometry [73].
- *CatCortex* is a brain network of cat, where nodes and links represent cortical regions and connections between them, the weight of a link denotes the densities of the connection [3].

Note that here we mainly focus on undirected networks, and thus we only extract the node, link, and weight information but ignore link direction. The basic topological features of the above six networks are shown in Table 1. In order to compare the results across different networks, all link weights are normalized to the interval $[0, 1]$ using

$$w^* = e^{-\frac{1}{w}}, \tag{24}$$

TABLE 1
Basic Topological Features of the Six Networks

	Celegans	USAir	Lesmis	NetScience	Geom	CatCortex
$\|V\|$	297	332	77	1,461	6,158	65
$\|E\|$	2,148	2,126	254	2,742	11,898	730
$\langle k \rangle$	14.465	12.807	6.597	3.754	3.864	22.46
C	0.308	0.749	0.736	0.878	0.728	0.667
$\langle d \rangle$	2.455	2.738	2.641	5.823	5.313	1.699
r	−0.163	−0.208	−0.165	0.462	0.243	−0.0283

$|V|$ and $|E|$ are the numbers of nodes and edges, respectively. $\langle k \rangle$ is the average degree. C is the clustering coefficient. $\langle d \rangle$ is the average distance and r denotes the assortativity coefficient.

where w and w^* denote the original and normalized weights, respectively [45].

4.2 Methods and Results

In this paper, we adopt several supervised learning algorithms such as *Random Forest* (RF) [75], *Gradient Boosting Decision Tree* (GBDT) [76], and *Support Vector Machine* [77] to make a crosswise comparison. To assess the prediction accuracy of our method, empirical experiments are conducted on the above six real-world networks. It should be noted that this study only focuses on link weight prediction. As part of the experiment, for each network, we randomly split its link weights into a training set W^T and a test set W^V, which contain 90 and 10 percent of the link weights, respectively, satisfying $W^T \cup W^V = W$ and $W^T \cap W^V = \emptyset$. Then, we treat all the 22 features mentioned in Section 3 as the input. By adopting RF, GBDT and SVM, we establish the models based on the training set, and then use them to predict the link weights in the test set, respectively. All the models are generated by R packages: *randomForest*, *gbm* and *e1071*. More specifically, in RF, we set *ntree* = 3,000; in GBDT, we set the *n.trees* = 3,000, *shrinkage* = 0.001 and *interaction.depth* = 2; in SVM, we choose the radial kernel and use grid search to find the suboptimal parameters, the values of *gamma* and *cost* are searched in the set [0.001, 0.01, 0.1, 1, 10, 100]. Comparing the weights estimated by the model and the actual ones in the test set, we calculate the *Pearson Correlation Coefficient* and *Root Mean Squared Error*. Repeating the process 30 times, we obtain the average PCC and RMSE, as shown in Tables 2 and 3, respectively. In our experiments, we also repeat the experiment for different times, ranging from 10 to 50, and find the corresponding results are quite similar.

We also compare our link weight prediction methods with the relatively new methods proposed in Ref. [44], Ref. [45] and Ref. [46]. In Ref. [44], Aicher et al. developed a weighted stochastic block model, which can be applied to infer both the existence and weights of links. Since we focus on link weight prediction, we choose pWSBM as one of our baselines. More specifically, we set 4 blocks and model the link weights with the normal distribution and exponential distribution, respectively. In Ref. [45], Zhao et al. proposed a generalize similarity indices based on reliable routes called rWCN, rWAA and rWRA. In Ref. [46], Zhu et al. proposed another method based on neighbor set. We treat all of them as the baseline methods. It should be noted that, in Zhao et al.'s and Zhu et al.'s work, they treated the link weight prediction as two different problems: the link prediction as a classification problem and the weight prediction as a regression problem by treating weights as link-existence probabilities. Most of those methods only used one or two local properties. In this work, however, we take link weight prediction as a regression problem using both original graph and line graph features. Furthermore, we also tried the *state-of-the-art* methods, i.e., *DeepWalk* [78] and *node2vec* [79], to automatically generate the feature vectors for nodes and links, and then use RF to realize link weight prediction. The dimension of embedding vectors achieved by *node2vec* is also set to 22, equal to the number of features in our models. Then the link representation can be acquired by hadamard operator [79]. In our experiments, the sampling parameter p is fixed to 1 and the optimal value of q is achieved through grid search from [0.125, 0.25, 0.5, 1, 2, 4, 8]. It should be noted that, the *DeepWalk* can be viewed as

TABLE 2
Prediction Accuracy under the Metric of PCC by Adopting Different Methods on Different Dataset, Using All 22 Features

Dataset	pWSBM	rWCN	rWAA	rWRA	Zhu et al.	node2vec+RF	RF	GBDT	SVM
Celegans	0.287	0.248	0.281	0.303	0.390	0.459	**0.502**	0.456	0.447
USAir	0.592	0.318	0.323	0.304	0.575	**0.777**	0.643	0.530	0.631
Lesmis	0.451	0.596	0.637	0.658	0.582	0.528	**0.720**	0.691	0.680
NetScience	0.539	0.381	0.478	0.493	0.293	0.639	**0.805**	0.776	0.770
Geom	0.491	0.463	0.488	0.394	0.494	0.564	**0.605**	0.551	0.512
CatCortex	0.405	0.331	0.362	0.409	0.229	0.476	0.473	0.470	**0.486**

TABLE 3
Prediction Accuracy under the Metric of RMSE by Adopting Different Methods on Different Dataset, Using All 22 Features

Dataset	pWSBM	rWCN	rWAA	rWRA	Zhu et al.	node2vec+RF	RF	GBDT	SVM
Celegans	0.209	0.429	0.443	0.478	0.204	0.190	**0.184**	0.189	0.192
USAir	0.00540	0.00587	0.00588	0.00593	0.00638	**0.00406**	0.00461	0.00499	0.00535
Lesmis	0.183	0.292	0.275	0.270	0.202	0.171	**0.140**	0.143	0.157
NetScience	0.121	0.157	0.142	0.138	0.169	0.114	**0.0875**	0.0929	0.0952
Geom	0.143	0.388	0.344	0.332	0.255	0.137	**0.131**	0.137	0.147
CatCortex	0.159	0.244	0.237	0.233	0.168	0.148	**0.147**	0.149	0.149

TABLE 4
Prediction Accuracy under the Metric of PCC by Using Different Groups of Features

Dataset	RF	RF (original graph)	RF (line graph)
Celegans	**0.502** (+7.73%)	0.466	0.459
USAir	**0.643** (+4.89%)	0.613	0.469
Lesmis	**0.720** (+2.13%)	0.705	0.608
NetScience	**0.805** (+1.90%)	0.790	0.741
Geom	**0.605** (+4.13%)	0.581	0.530
CatCortex	**0.473** (+4.67%)	0.452	0.407

TABLE 5
Prediction Accuracy under the Metric of RMSE by Using Different Groups of Features

Dataset	RF	RF (original graph)	RF (line graph)
Celegans	**0.184** (+2.13%)	0.188	0.189
USAir	**0.00461** (+4.16%)	0.00481	0.00555
Lesmis	**0.140** (+4.76%)	0.147	0.158
NetScience	**0.0875** (+4.58%)	0.0917	0.0967
Geom	**0.131** (+2.24%)	0.134	0.139
CatCortex	**0.147** (+2.65%)	0.151	0.154

the special case of *node2vec* when the sampling parameters set to be $p = q = 1$.

As shown in Tables 2 and 3, we obtain larger PCC but smaller RMSE by adopting any supervised learning method than the baseline methods on most datasets, indicating that supervised learning methods indeed perform much better than the others using any metric. For PCC, all the p-values are smaller than 0.0001, suggesting the statistically significant positive linear correlation between the weights estimated by the supervised learning methods and the real ones. By comparison, the decision tree based algorithms, i.e., RF and GBDT, perform better than SVM. Moreover, we find our features perform better than the features automatically generated by *node2vec* in most cases, by using the same supervised learning algorithm RF.

It is known that the original graph features, such as similarity indices, are widely used in link prediction. However, for link weight prediction, the weights in many networks are not only correlated with the similarity, e.g., when the frequency is viewed as the link weight, thus the similarity as one indicator may not represent all the information about link weight. Furthermore, most original graph features are calculated by pairs of nodes, and ignore the information of link itself. In our work, we extract the centrality indices in line graph, which can be directly viewed as link information. In Tables 4 and 5 we compare the RMSE or PCC results of RF algorithm by using original graph features, line graph features and both, respectively. We find that the results using both groups of features together can indeed improve the performances, in terms of larger PCC and smaller RMSE. That is, the line graph features are complementary to original graph features in link weight prediction.

In order to address the robustness of supervised learning methods on the size of training set, we obtain the prediction accuracies for the link weight prediction using various sizes of training sets (from 10 to 90 percent with 20 percent interval). For each size, we randomly divide the training set and test set for 30 times and record the average result. The results are shown in Figs. 2 and 3 for the metrics of PCC and RMSE, respectively. We find that the results obtained

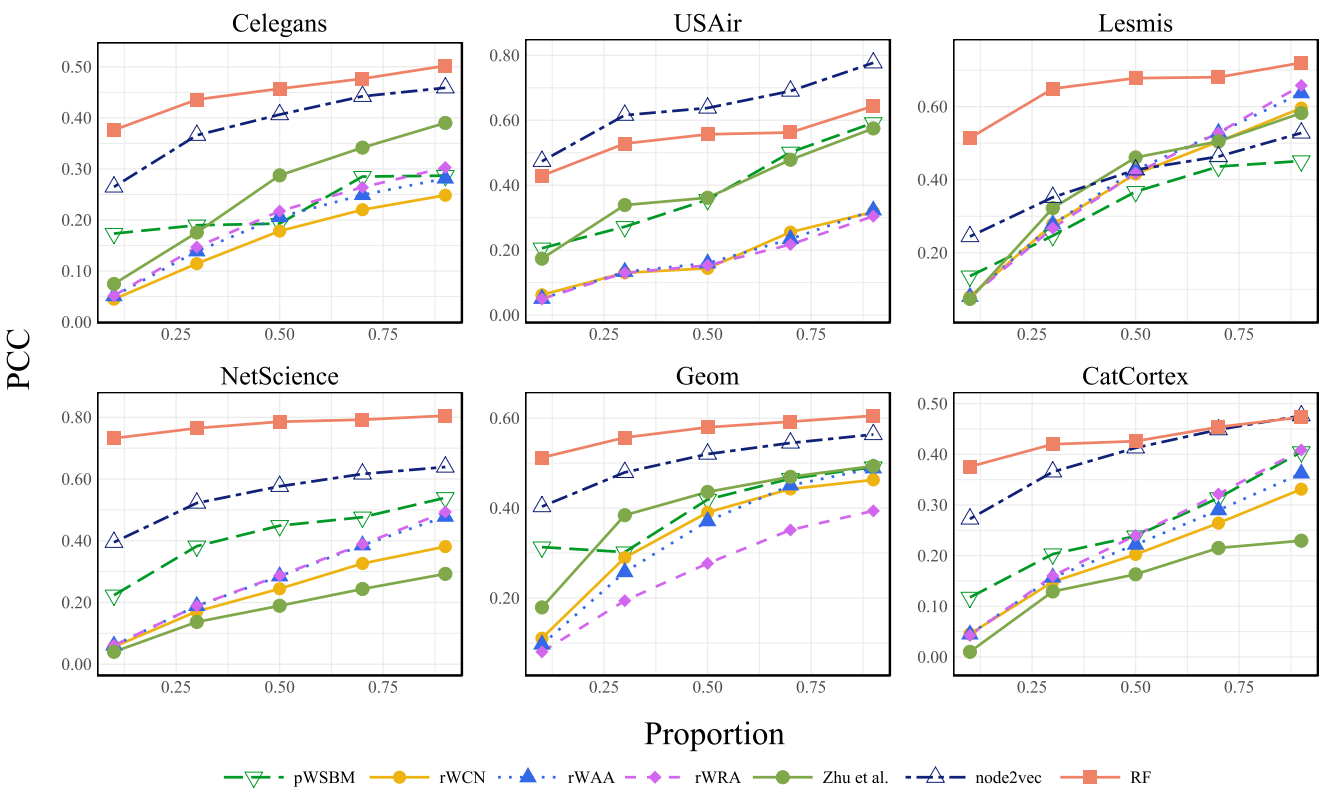


Fig. 2. The metric of PCC as functions of the size of training set (represented by the fraction of samples in the training set), for different methods.

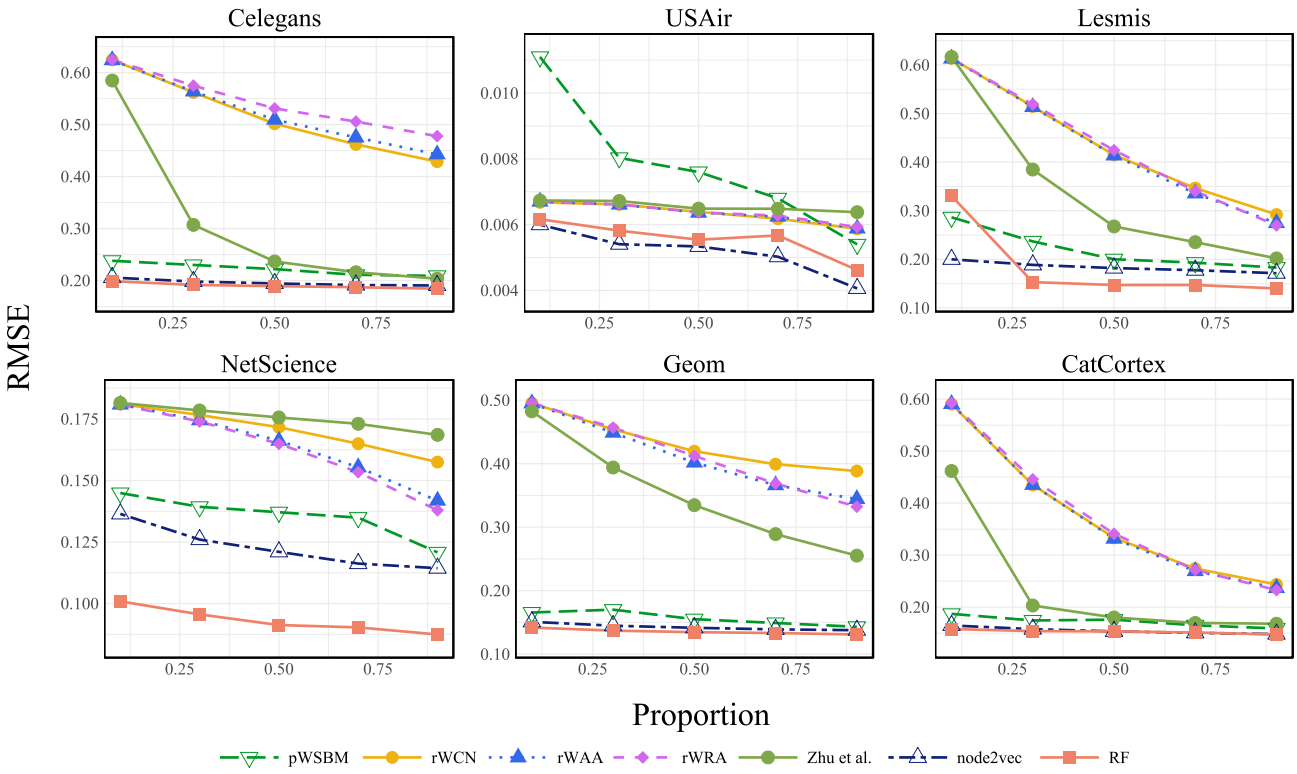


Fig. 3. The metric of RMSE as functions of the size of training set (represented by the fraction of samples in the training set), for different methods.

by our method using RF are better than those obtained by the baseline methods in most cases, indicating the robustness of our method on the size of training set.

Besides, we also investigate the feature importance based on the Random Forest mean decreasing accuracy and mean decreasing impurity, as shown in Tables 6 and 7, respectively, with all the values normalized. We underline the top 3 features in each dataset. Relatively speaking, RA is the most important feature in our study, which is consistent with the previous work [30], while in line graph, we cannot find a dominant feature that behaves better than the others in most cases, i.e., BC, EC, PR have their advantages in different networks, respectively.

4.3 Time Complexity

Generally, calculating the node centrality is time consuming. In this part, we study the time complexity of feature indices in our methods. Since our centrality

TABLE 6
Feature Importance Determined by the Random Forest Normalized Mean Decreasing Accuracy Measure

Metrics	Celegans	USAir	Lesmis	NetScience	Geom	CatCortex
CN	0.224	0.443	0.383	0.384	0.199	0.329
SA	0.546	0.389	0.348	0.413	0.244	**0.937**
JAC	0.471	0.312	0.305	0.387	0.230	0.776
HPI	**0.701**	0.611	0.332	0.461	0.271	**1.000**
HDI	0.456	0.378	0.330	0.319	0.225	0.739
SI	0.460	0.335	0.301	0.320	0.208	0.797
LHN	0.546	0.432	0.343	0.287	0.238	0.825
AA	0.583	0.518	**0.593**	0.472	**0.367**	0.558
RA	**1.000**	0.714	**1.000**	**1.000**	**1.000**	**0.977**
PA	0.568	0.471	0.248	0.290	0.233	0.573
FM	0.672	0.369	**0.406**	0.385	0.246	0.574
LP	0.398	**0.768**	0.230	0.229	0.085	0.459
LRW	0.684	0.352	0.225	0.446	0.285	0.665
EB	0.556	**0.772**	0.361	0.285	0.301	0.581
DC	0.634	0.269	0.184	0.209	0.162	0.474
CC	0.604	0.505	0.298	**0.496**	0.281	0.921
BC	0.660	**1.000**	0.205	0.327	0.201	0.895
EC	0.670	0.276	0.330	0.352	0.298	0.769
PR	**0.688**	0.264	0.298	**0.549**	**0.384**	0.544
C	0.365	0.236	0.363	0.284	0.297	0.480
H	0.651	0.325	0.276	0.301	0.145	0.375
CO	0.468	0.265	0.285	0.353	0.188	0.406

TABLE 7
Feature Importance Determined by the Random Forest Normalized Mean Decreasing Impurity Measure

Metrics	Celegans	USAir	Lesmis	NetScience	Geom	CatCortex
CN	0.091	0.128	**0.226**	0.210	0.127	0.132
SA	0.498	0.149	0.147	**0.368**	0.183	**1.000**
JAC	0.360	0.144	0.127	0.331	0.153	0.669
HPI	0.551	0.265	0.142	**0.421**	0.210	**0.987**
HDI	0.358	0.152	0.106	0.322	0.143	0.722
SI	0.336	0.144	0.110	0.319	0.140	0.688
LHN	0.487	0.208	0.131	0.365	0.142	0.712
AA	0.525	0.300	**0.450**	0.254	0.353	0.428
RA	**0.932**	**0.359**	**1.000**	**1.000**	**1.000**	**0.843**
PA	0.462	0.153	0.081	0.077	**0.516**	0.338
FM	0.598	0.239	0.205	0.106	0.294	0.458
LP	0.455	0.319	0.120	0.060	0.137	0.821
LRW	**0.834**	0.143	0.115	0.200	0.358	0.566
EB	0.689	**1.000**	0.128	0.100	0.297	0.648
DC	0.647	0.107	0.053	0.057	0.277	0.228
CC	0.756	0.168	0.107	0.182	0.433	0.599
BC	0.694	**0.493**	0.114	0.118	0.281	0.611
EC	**1.000**	0.181	0.135	0.083	**0.507**	0.539
PR	0.805	0.148	0.109	0.290	0.350	0.434
C	0.674	0.167	0.185	0.082	0.193	0.573
H	0.695	0.086	0.066	0.054	0.152	0.227
CO	0.445	0.089	0.058	0.058	0.134	0.205

TABLE 8
The Computation Complexity of Similarity and Centrality Indices in Original Graph and Line Graph, Respectively

Metrics	Original graph	Line graph
CN	$O(\|V\|)$	-
SA	$O(\|V\|)$	-
JAC	$O(\|V\|)$	-
HPI	$O(\|V\|)$	-
HDI	$O(\|V\|)$	-
SI	$O(\|V\|)$	-
LHN	$O(\|V\|)$	-
AA	$O(\|V\|)$	-
RA	$O(\|V\|)$	-
PA	$O(\|V\|)$	-
FM	$\approx O(\|V\|\langle k\rangle)$	-
LP	$\approx O(\|V\|\langle k\rangle)$	-
LRW	$O(\|V\|S)$	-
EB	$O(\|V\|^3)$	-
DC	$O(\|V\|)$	$O(\|E\|)$
CC	$O(\|V\|^3)$	$O(\|E\|^3)$
BC	$O(\|V\|^3)$	$O(\|E\|^3)$
EC	$O(\|V\|^2)$	$O(\|E\|^2)$
PR	$O(\|E\|S)$	$\approx O((\|V\|\langle k\rangle^2 - \|E\|)S)$
C	$\approx O(\|V\|\langle k\rangle)$	$\approx O(\|E\|\langle k\rangle)$
H	$\approx O(\|V\|\langle k\rangle)$	$\approx O(\|E\|\langle k\rangle)$
CO	$O(\|E\|)$	$\approx O(\|V\|\langle k\rangle^2 - \|E\|)$

indices are calculated in line graph, we should calculate the basic characteristics in line graph first. Assuming that the original graph has $|V|$ nodes and $|E|$ edges, then the number of nodes in line graph is equal to $|E|$ and the number of edges is equal to $\sum_{i=1}^{|V|}(k_i)^2 - |E|$ [48]. Thus we can easily transform the time complexity of original graph to line graph, e.g., the time complexity of degree centrality in original graph is $O(|V|)$, then its time complexity in line graph is $O(|E|)$. More time complexity of centrality indices and similarity indices are shown in Table 8, where $\langle k\rangle$ denotes average degree in original graph and S denotes the steps of iterations of corresponding algorithm. Moreover, for unsupervised methods in our work, the time complexity are almost the same as the feature extraction. For supervised methods,the time complexity is mainly determined by the training part, e.g., the time complexity for SVM is $O(|E|^3)$ [80], for RF and GBDT is $O(TF|E|\log|E|)$, where T denotes the number of trees, F denotes the number of attributes [81].

5 Application on Yelp Layered Network

Although many real-world systems can be roughly described by single-layer networks, some of them contain different kinds of nodes and links, and thus the concepts such as network of networks and layered network emerge and attract lots of attentions from network researchers [19], [42], [82], [83], [84]. A layered network can be used to describe a system with different kinds of relationships. For example, in *Open Source Software* projects, there are two kinds of relationships between developers: the social relationship by emailing each other and the collaboration relationship by working together [42].

Here, we focus on Yelp network. Yelp is a popular website for reviewing restaurants, stores and so on. Our study is based on the recently released Yelp Dataset Challenge.[1] The dataset contains information about user ID, reviews, business attributes, and so on. Yelp network is a layered network because it contains different kinds of relationships. By using the information about user ID, friends and review history on restaurants, we can construct a network of two layers. In the first layer, two users are connected if they are friends, namely social network. In the second layer, two users are connected if they reviewed the restaurants in at least one same cluster, namely co-foraging network. Here the cluster of restaurants is based on the restaurant locations and is realized by using Density-Based Spatial Clustering of Application with Noise (DBSCAN) method [85]. The link in the co-foraging network has a weight, defined as the number of clusters of restaurants that the corresponding two users have ever reviewed. In this layered network, we only focus on the those active users with more than 50 reviews, and finally the Yelp layered network totally contains 2121 nodes and 35520 social links. Since we want to evaluate the effect of social links on the co-foraging behavior of the pairwise users, only the pairwise users connected by social links are considered here, while those pairs of users with no social link between them are ignored.

Based on this layered network, we extract the 22 features for each link in the social network, take them as the input, and treat the corresponding link weight in the co-foraging network as the output. Then, we divide this dataset into training set and test set, containing 90 and 10 percent samples, respectively. Due to the better performance of RF in the previous experiments, we use them to establish the prediction model for the Yelp layered network to realize cross-layer link weight prediction. The comparisons of PCC and RMSE are shown in Table 9, where we can see that RF using our handcrafted features is comparable with that using the features automatically generated by *node2vec*, both of which performers much better than the unsupervised baseline methods, and this result is quite stable for various size of training set, as shown in Fig. 4. These imply that the online social interactions can indeed be used to predict the offline co-foraging behavior to certain extent, by using supervised learning methods.

6 Conclusion

In this paper, we adopted supervised learning methods, such as RF, GBDT, and SVM, by utilizing the features including similarity indices in original graph and centrality indices in line graph to realize the link weight prediction in various networks. Specifically, we also apply them in Yelp layered network to realize cross-layer link weight prediction. The results showed that, our supervised learning methods can get much better prediction performance, in terms of larger PCC and smaller RMSE, than baseline methods, and such superiority is quite robust for various sizes of training sets. Moreover, we also find that the RA plays a

1. https://www.yelp.com/dataset_challenge

TABLE 9
The Metrics of PCC and RMSE by Adopting Different Methods on Yelp Dataset

Metrics	pWSBM	rWCN	rWAA	rWRA	Zhu et al.	node2vec+RF	RF	RF (original graph)	RF (line graph)
PCC	0.561	0.250	0.238	0.170	0.310	**0.822**	0.761	0.712	0.742
RMSE	0.124	0.616	0.631	0.717	0.171	**0.0812**	0.0913	0.0992	0.0951

more important role in the weight prediction, which is consistent with Zhou et al.'s work [30].

The study on Yelp layered network suggests that our methods can also be used to predict the offline co-foraging behavior of users just based on their online social interactions, which may open a new direction for link prediction algorithms, and meanwhile provide insights to design better restaurant recommendation system. Our studies highlight the fact that supervised learning methods, assistant with appropriate network properties, can achieve great success in link weight prediction. It should be noted that the current algorithms, such as *node2vec*, can only generate the feature vectors for nodes, not for links directly. It is naturally to believe that, in this way, some important link information could be ignored by using this method. Therefore, we are trying to create a new algorithm, namely link2vec, based on our line graph to automatically generate the feature vectors for links, and then use supervised learning methods to realize link weight prediction, which we think can further improve the link weight prediction performance and belongs to our future work. Furthermore, the link weight prediction is similar to the link prediction, and thus has two meanings: one is to uncover the missing weight of links, another is to predict the future weight of interactions. In our present work, we mainly focus on the first meaning, but time is certainly a very important factor in the real-world systems and thus should be addressed. Therefore, in our future work, we will take the temporal network into consideration, collect more temporal weighted networks, and further abstract significant temporal features to predict the temporal weight of links.

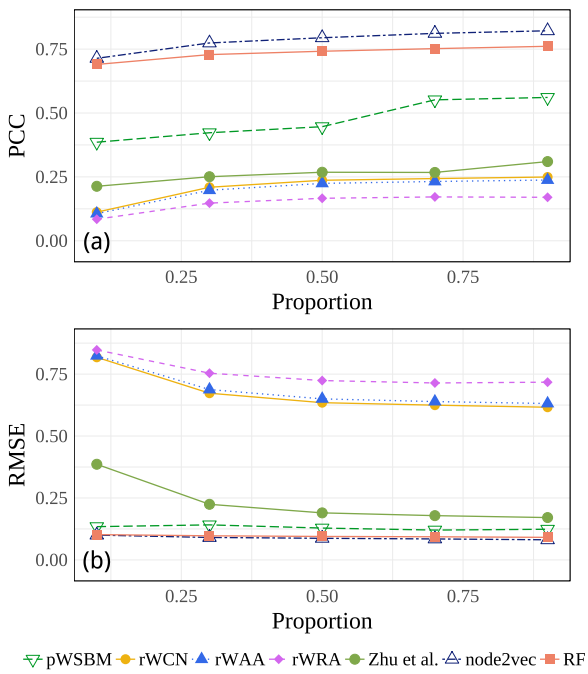


Fig. 4. The metrics of PCC and RMSE as functions of the size of training set sizes (represented by the fraction of samples in the training set), for different methods on Yelp layered network.

Acknowledgments

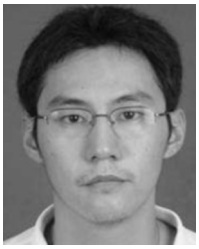
This work is partially supported by the National Natural Science Foundation of China (11505153, 61572439, 61502423, 61273212, 61573310) and the Natural Science Foundation of Zhejiang Province (LQ15A050002, LY18F010025, LY15F030006).

References

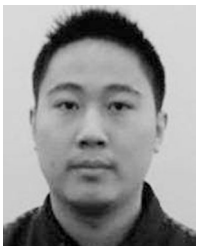
[1] M. E. J. Newman, "The structure and function of complex networks," *SIAM Rev.*, vol. 45, no. 2, pp. 167–256, 2003.
[2] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. U. Hwang, "Complex networks: Structure and dynamics," *Physics Rep.*, vol. 424, no. 4/5, pp. 175–308, 2006.
[3] J. W. Scannell, C. Blakemore, and M. P. Young, "Analysis of connectivity in the cat cerebral cortex," *J. Neurosci.*, vol. 15, no. 2, pp. 1463–1483, 1995.
[4] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, no. 6825, pp. 268–276, 2001.
[5] Q. Xuan, H. Fang, C. Fu, and V. Filkov, "Temporal motifs reveal collaboration patterns in online task-oriented networks," *Phys. Rev. E*, vol. 91, no. 5, 2015, Art. no. 052813.
[6] C. Fu, J. Wang, Y. Xiang, Z. Wu, L. Yu, and Q. Xuan, "Pinning control of clustered complex networks with different size," *Physica A: Statistical Mech. Appl.*, vol. 479, pp. 184–192, 2017.
[7] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, no. 6833, pp. 41–42, 2001.
[8] L. D. F. Costa, F. A. Rodrigues, G. Travieso, and P. R. Villas Boas, "Characterization of complex networks: A survey of measurements," *Advances Physics*, vol. 56, no. 1, pp. 167–242, 2007.
[9] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
[10] M. E. J. Newman, "Assortative mixing in networks," *Phys. Rev. Lett.*, vol. 89, no. 20, 2002, Art. no. 208701.
[11] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mech. Appl.*, vol. 390, no. 6, pp. 1150–1170, 2011.
[12] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Sci.*, vol. 286, no. 5439, pp. 509–512, 1999.
[13] Q. Xuan, Y. Li, and T. J. Wu, "Growth model for complex networks with hierarchical and modular structures," *Phys. Rev. E*, vol. 73, no. 3, 2006, Art. no. 036105.
[14] Q. Xuan, C. Fu, and L. Yu, "Ranking developer candidates by social links," *Advances Complex Syst.*, vol. 17, no. 7n08, 2014, Art. no. 1550005.
[15] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web," *Stanford Digital Libraries Work. Paper*, vol. 9, no. 1, pp. 1–14, 1998.
[16] J. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "The web as a graph: Measurements, models, and methods," in *Proc. Int. Comput. Combinatorics Conf.*, 1999, pp. 1–17.
[17] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *J. Assoc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, 2007.
[18] Y. Pan, D.-H. Li, J.-G. Liu, and J.-Z. Liang, "Detecting community structure in complex networks via node similarity," *Physica A: Statistical Mech. Appl.*, vol. 389, no. 14, pp. 2849–2857, 2010.
[19] Q. Xuan and T. J. Wu, "Node matching between complex networks," *Phys. Rev. E*, vol. 80, no. 2, 2009, Art. no. 026103.

[20] Q. Xuan, F. Du, and T.-J. Wu, "Iterative node matching between complex networks," *J. Physics A: Math. Theoretical*, vol. 43, no. 39, 2010, Art. no. 395002.
[21] P. Holme and J. Saramki, "Temporal networks," *Physics Rep.*, vol. 519, no. 3, pp. 97–125, 2011.
[22] N. Masuda and P. Holme, "Introduction to temporal network epidemiology," in *Temporal Network Epidemiology*. Berlin, Germany: Springer, 2017, pp. 1–16.
[23] X.-W. Chen and M. Liu, "Prediction of protein–protein interactions using random decision forest framework," *Bioinf.*, vol. 21, no. 24, pp. 4394–4400, 2005.
[24] J. Yang, T. Yang, D. Wu, L. Lin, F. Yang, and J. Zhao, "The integration of weighted human gene association networks based on link prediction," *BMC Syst. Biol.*, vol. 11, no. 1, 2017, Art. no. 12.
[25] L. Lü, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, and T. Zhou, "Recommender systems," *Physics Rep.*, vol. 519, no. 1, pp. 1–49, 2012.
[26] P. Resnick and H. R. Varian, "Recommender systems," *Commun. ACM*, vol. 40, no. 3, pp. 56–58, 1997.
[27] H. Chen, X. Li, and Z. Huang, "Link prediction approach to collaborative filtering," in *Proc. 5th ACM/IEEE-CS Joint Conf. Digit. Libraries*, 2005, pp. 141–142.
[28] H. H. Song, T. W. Cho, V. Dave, Y. Zhang, and L. Qiu, "Scalable proximity estimation and link prediction in online social networks," in *Proc. 9th ACM SIGCOMM Conf. Internet Meas. Conf.*, 2009, pp. 322–335.
[29] W. Cukierski, B. Hamner, and B. Yang, "Graph-based features for supervised link prediction," in *Proc. Int. Joint Conf. Neural Netw.*, 2011, pp. 1237–1244.
[30] T. Zhou, L. Lü, and Y.-C. Zhang, "Predicting missing links via local information," *Eur. Phys. J. B-Condensed Matter Complex Syst.*, vol. 71, no. 4, pp. 623–630, 2009.
[31] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *Proc. Workshop Link Anal. Counter-Terrorism Secur.*, 2006, http://www.siam.org/meetings/sdm06/workproceed/Link%20Analysis/index.html
[32] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla, "New perspectives and methods in link prediction," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 243–252.
[33] S. Scellato, A. Noulas, and C. Mascolo, "Exploiting place features in link prediction on location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 1046–1054.
[34] M. E. J. Newman, "Analysis of weighted networks," *Phys. Rev. E*, vol. 70, no. 5, 2004, Art. no. 056131.
[35] J.-P. Onnela, J. Saramäki, J. Kertész, and K. Kaski, "Intensity and coherence of motifs in weighted complex networks," *Phys. Rev. E*, vol. 71, no. 6, 2005, Art. no. 065103.
[36] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis," *Statistical Appl. Genetics Molecular Biol.*, vol. 4, no. 1, 2005, Art. no. 1128.
[37] W. Li and X. Cai, "Statistical analysis of airport network of china," *Phys. Rev. E*, vol. 69, no. 4, 2004, Art. no. 046106.
[38] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," *Proc. Nat. Academy Sci. USA*, vol. 106, no. 36, pp. 15 274–15 278, 2009.
[39] T. Murata and S. Moriyasu, "Link prediction of social networks based on weighted proximity measures," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, 2007, pp. 85–88.
[40] L. Lü and T. Zhou, "Link prediction in weighted networks: The role of weak ties," *Europhysics Lett.*, vol. 89, no. 1, 2010, Art. no. 18001.
[41] L. Backstrom and J. Kleinberg, "Romantic partnerships and the dispersion of social ties: A network analysis of relationship status on Facebook," in *Proc. 17th ACM Conf. Comput. Supported Cooperative Work Social Comput.*, 2014, pp. 831–841.
[42] Q. Xuan and V. Filkov, "Building it together: Synchronous development in OSS," in *Proc. 36th Int. Conf. Softw. Eng.*, 2014, pp. 222–233.
[43] V. S. Vijayaraghavan, P.-A. Noël, Z. Maoz, and R. M. DSouza, "Quantifying dynamical spillover in co-evolving multiplex networks," *Sci. Rep.*, vol. 5, 2015, Art. no. 15142.
[44] C. Aicher, A. Z. Jacobs, and A. Clauset, "Learning latent block structure in weighted networks," *J. Complex Netw.*, vol. 3, no. 2, pp. 221–248, 2014.
[45] J. Zhao, et al., "Prediction of links and weights in networks by reliable routes," *Sci. Rep.*, vol. 5, 2015, Art. no. 12261.
[46] B. Zhu, Y. Xia, and X. J. Zhang, "Weight prediction in complex networks based on neighbor set," *Sci. Rep.*, vol. 6, 2016, Art. no. 38080.
[47] H. R. De Sá and R. B. Prudêncio, "Supervised link prediction in weighted networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2011, pp. 2281–2288.
[48] F. Harary and R. Z. Norman, "Some properties of line digraphs," *Rendiconti Del Circolo Matematico di Palermo*, vol. 9, no. 2, pp. 161–168, 1960.
[49] G. Kossinets and D. J. Watts, "Empirical analysis of an evolving social network," *Sci.*, vol. 311, no. 5757, pp. 88–90, 2006.
[50] D. Lin, "An information-theoretic definition of similarity," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 296–304.
[51] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Nat. Academy Sci. USA*, vol. 99, no. 12, pp. 7821–7826, 2002.
[52] M. E. J. Newman, "Clustering and preferential attachment in growing networks," *Phys. Rev. E*, vol. 64, no. 2, 2001, Art. no. 025102.
[53] G. Kossinets, "Effects of missing data in social networks," *Social Netw.*, vol. 28, no. 3, pp. 247–268, 2006.
[54] G. Salton and M. J. Mcgill, *Introduction to Modern Information Retrieval*. London, U.K.: Facet Publishing, 2010.
[55] P. Jaccard, "Étude de la distribution florale dans une portion des alpes et du jura," *Bulletin De La Societe Vaudoise Des Sciences Naturelles*, vol. 37, no. 142, pp. 547–579, 1901.
[56] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabsi, "Hierarchical organization of modularity in metabolic networks," *Sci.*, vol. 297, no. 5586, 2002, Art. no. 1551.
[57] T. Sørensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons," *Biologiske Skrifter*, vol. 5, pp. 1–34, 1948.
[58] E. A. Leicht, P. Holme, and M. E. J. Newman, "Vertex similarity in networks," *Phys. Rev. E*, vol. 73, no. 2, 2005, Art. no. 026120.
[59] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Netw.*, vol. 25, no. 3, pp. 211–230, 2003.
[60] Q. Ou, Y.-D. Jin, T. Zhou, B.-H. Wang, and B.-Q. Yin, "Power-law strength-degree correlation from resource-allocation dynamics on weighted networks," *Phys. Rev. E*, vol. 75, no. 2, 2007, Art. no. 021102.
[61] M. Fire, L. Tenenboim, O. Lesser, R. Puzis, L. Rokach, and Y. Elovici, "Link prediction in social networks using computationally efficient topological features," in *Proc. IEEE 3rd Int. Conf. Privacy Secur. Risk Trust*, 2011, pp. 73–80.
[62] W. Liu and L. Lü, "Link prediction based on local random walk," *Europhysics Lett.*, vol. 89, no. 5, pp. 58 007–58 012(6), 2010.
[63] J. G. Liu, Z. M. Ren, Q. Guo, and B. H. Wang, "Node importance ranking of complex networks," *Acta Physica Sinica*, vol. 62, no. 17, 2013, Art. no. 178901.
[64] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Netw.*, vol. 1, no. 3, pp. 215–239, 1978.
[65] A. Bavelas, "Communication patterns in task-oriented groups," *J. Acoustical Soc. America*, vol. 22, no. 6, pp. 725–730, 2005.
[66] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977.
[67] P. Bonacich, "Power and centrality: A family of measures," *Amer. J. Sociology*, vol. 92, no. 5, pp. 1170–1182, 1987.
[68] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proc. Nat. Academy Sci. USA*, vol. 102, no. 46, pp. 16 569–16 572, 2005.
[69] L. Lü, T. Zhou, Q.-M. Zhang, and H. E. Stanley, "The H-index of a network node and its relation to degree and coreness," *Nature Commun.*, vol. 7, 2016, Art. no. 10168.
[70] S. B. Seidman, "Network structure and minimum degree," *Social Netw.*, vol. 5, no. 3, pp. 269–287, 1983.
[71] V. Batagelj and M. Zaversnik, "An O(m) algorithm for cores decomposition of networks," *Comput. Sci.*, vol. 1, no. 6, pp. 34–37, 2003.
[72] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir, "A model of internet topology using k-shell decomposition," *Proc. Nat. Academy Sci. USA*, vol. 104, no. 27, pp. 11 150–11 154, 2007.
[73] V. Batagelj and A. Mrvar, "Pajek datasets [ol]," 2006. [Online]. Available: http://vlado.fmf.uni-lj.si/pub/networks/data/
[74] D. E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing*, vol. 37. Reading, MA, USA: Addison-Wesley, 1993.
[75] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[76] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, pp. 1189–1232, 2001.
[77] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 1997, pp. 155–161.

[78] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 701–710.
[79] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 855–864.
[80] A. Abdiansah and R. Wardoyo, "Time complexity analysis of support vector machines (SVM) in LibSVM," *Int. J. Comput. Appl.*, vol. 128, no. 3, pp. 0975–8887, 2015.
[81] G. Biau, "Analysis of a random forests model," *J. Mach. Learn. Res.*, vol. 13, pp. 1063–1095, 2012.
[82] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *J. Complex Netw.*, vol. 2, no. 3, pp. 203–271, 2014.
[83] S. Gomez, A. Diaz-Guilera, J. Gomez-Gardenes, C. J. Perez-Vicente, Y. Moreno, and A. Arenas, "Diffusion dynamics on multiplex networks," *Phys. Rev. Lett.*, vol. 110, no. 2, 2013, Art. no. 028701.
[84] D. Y. Kenett, M. Perc, and S. Boccaletti, "Networks of networks—An introduction," *Chaos Solitons Fractals*, vol. 80, pp. 1–6, 2015.
[85] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.



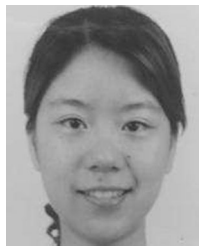
Chenbo Fu received the BS degree in physics from the Zhejiang University of Technology, in 2007, and the MS and PhD degrees in physics from Zhejiang University, in 2009 and 2013, respectively. He was a postdoctoral researcher in the College of Information Engineering, Zhejiang University of Technology, and was a research assistant in the Department of Computer Science, University of California at Davis, in 2014. Currently, he is lecturer in the College of Information Engineering, Zhejiang University of Technology. His research interests including network based algorithm design, social network data mining, chaos synchronization, network dynamics, and machine learning.



Minghao Zhao received the BS degree from the Wuhan University of Technology, China, in 2014. He is working toward the MS degree in control theory and engineering at the Zhejiang University of Technology. His research interests include social network analysis and machine learning.

Lu Fan received the BS degree from the Zhejiang University of Technology, China, in 2017. Her research interests include recommender systems, spatial-temporal data mining, and graph mining.

Xinyi Chen received the BS degree from the Zhejiang University of Technology, China, in 2017. His research interests include graph mining and machine learning.



Jinyin Chen received the BS and PhD degrees from the Zhejiang University of Technology, Hangzhou, China, in 2004 and 2009, respectively. She is currently an associate professor in the College of Information Engineering, Zhejiang University of Technology, China. Her research interests include cover intelligent computing, optimization, and network security.

Zhefu Wu received the PhD degree from the College of Information Science & Electronic Engineering, Zhejiang University, Hangzhou, China, in 2000. He is currently an associate professor in the College of Information Engineering, Zhejiang University of Technology, China. He has special research interests which includes social network data mining, complex network dynamics, machine learning, wireless sensor network algorithms, and applications.

Yongxiang Xia received the BS and PhD degrees in electronic engineering both from Tsinghua University, China, in 1998 and 2004, respectively. From 2004 to 2006, he was a postdoctoral fellow with the Hong Kong Polytechnic University. After that, he worked at the Australian National University as a research fellow. He joined Zhejiang University in 2010 and currently is an associate professor. His current research is in the area of network science and its application in engineering networks, where he has published more than 40 papers. He is a senior member of the IEEE, a member of the IEEE Technical Committee on Nonlinear Circuits and Systems, an associate editor of the *IEEE Transactions on Circuits and Systems-II: Express Briefs*, and an editorial board member of the *Scientific Reports*.

Qi Xuan received the BS and PhD degrees in control theory and engineering from Zhejiang University, Hangzhou, China, in 2003 and 2008, respectively. He was a post-doctoral researcher in the Department of Information Science and Electronic Engineering, Zhejiang University, from 2008 to 2010, and a research assistant in the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, China, in 2010. From 2012 to 2014, he was a postdoctoral fellow in the Department of Computer Science, University of California at Davis, Davis, California. He is currently a professor in the College of Information Engineering, Zhejiang University of Technology, Hangzhou. His current research interests include network-based algorithm design, social network data mining, social synchronization and consensus, reaction-diffusion network dynamics, machine learning, and computer vision.