Expert Systems With Applications 165 (2021) 113896

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A hybrid method of link prediction in directed graphs

Hossien Ghorbanzadeh [a], Amir Sheikhahmadi [a,*], Mahdi Jalili [b], Sadegh Sulaimany [c]

[a] Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran
[b] School of Engineering, RMIT University, Melbourne, Australia
[c] Department of Computer Engineering, Faculty of Engineering, University of Kurdistan, Sanandaj, Iran

ARTICLE INFO

Keywords:
Link Prediction
Structural Similarity
Local Similarity Measure
Common Neighborhood
Supervised Learning
Unsupervised Learning

ABSTRACT

Link prediction is an important issue in complex network analysis and mining. Given the structure of a network, a link prediction algorithm obtains the probability that a link is established between two non-adjacent nodes in the future snapshots of the network. Many of the available link prediction methods are based on common neighborhood. A problem with these methods is that if two nodes do not have any common neighbors, they always predict a chance of zero for establishment of a link between them; however, such nodes have been shown to establish links in some real systems. Another issue with these measures is that they often disregard the connection direction. Here, we propose a novel measure based on common neighborhood that resolves the above issues. The proposed measures are applied on three benchmark networks in both unsupervised and supervised learning modes. Our experiments show the superior performance of the proposed measures over that of the state-of-the-art link prediction methods.

1. Introduction

An online social network reflects the reality of human communities, and contains a large amount of information about them, a phenomenon that has led many researchers in various fields to analyze their features. A social network includes a set of social actors and the relationships between them. It can be represented as a graph, where the nodes represent the actors, and the edges indicate the links between them. Relationships are usually established between people because of their common tendencies. Addition of new nodes and relationships to a social network creates dynamic behavior. It is important in the context of social network analysis and mining to predict the probability that a relationship is established between two nodes. This is widely known as the link prediction problem. If there is an instant image of a social network at time t, the purpose of link prediction is to predict the edges that are likely to be established in the period between t and $t + 1$ ($t < t + 1$) (Liben-Nowell and Kleinberg, 2007). The applications of link prediction in online social networks include recommendation of friends to new members and movie and music recommendation systems based on users' backgrounds.

Various methods have been proposed for solution of link prediction problems in network systems. The structural similarities between nodes on the network graph are used in some of these methods. Recently, measures based on local similarity have received much attention for solving the problem of link prediction. This is partly due to the simplicity and yet effectiveness of these measures and their relatively proper prediction accuracy in social networks. Most of them use node degree and features based on the common neighbors, disregarding the relations between common neighbors (Martínez, Berzal, & Cubero, 2016). Moreover, conventional local measures, such as common neighbors, Adamic Adar, and Resource Allocation, fail to consider the directions of neighborhood. In fact, such measures make no distinction between directed and undirected graphs. Aghabozorgi and Khayyambashi (2018) presented a local measure that uses not only the number of common neighbors but also triadic structure blocks to obtain the similarity between two nodes. The neighborhood directions of pairs of nodes are not considered in this measure either, where only triadic blocks are examined while larger ones are not assessed. Furthermore, this method is based on network motifs, which has high computational complexity and is not practical for large graphs. Z. Li, Fang, and Sheng (2017) investigated link recommendation systems. (Pecli, Cavalcanti, & Goldschmidt, 2018) presented a supervised method of link prediction. They demonstrated that their feature selection method improves the efficiency of classifiers. S. Li, et al. (2018) presented a similarity measure based on

* Corresponding author.
E-mail addresses: hossein.ghorbanzadeh@iauahvaz.ac.ir (H. Ghorbanzadeh), asheikhahmadi@iausdj.ac.ir (A. Sheikhahmadi), Mahdi.jalili@rmit.edu.au (M. Jalili), S.Sulaimany@uok.ac.ir (S. Sulaimany).

https://doi.org/10.1016/j.eswa.2020.113896
Received 18 February 2020; Received in revised form 29 June 2020; Accepted 16 August 2020
Available online 31 August 2020

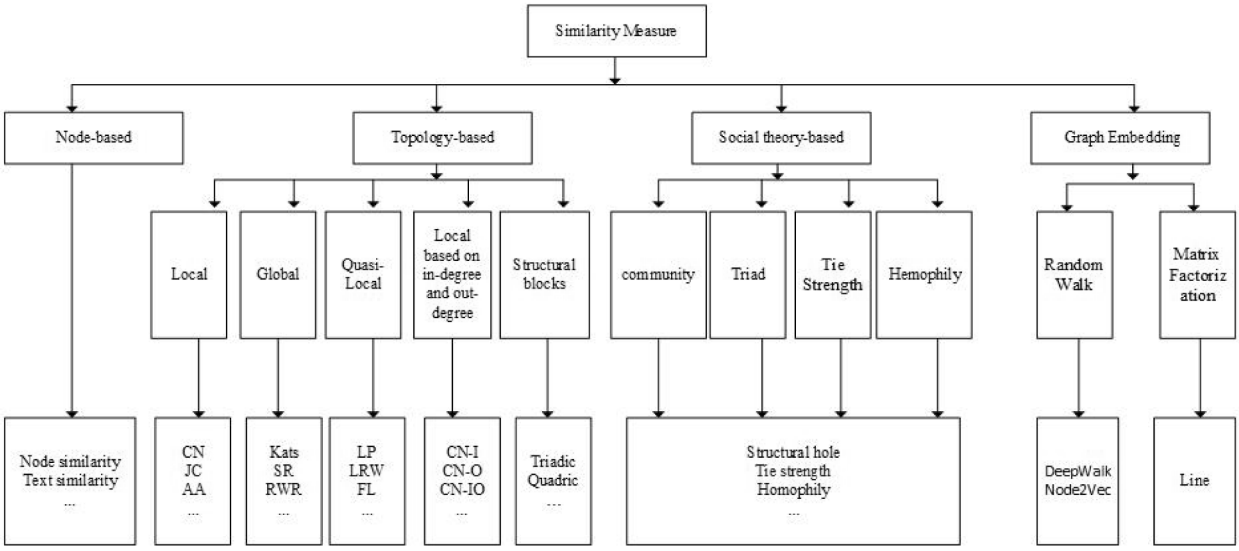


Fig. 1. A classification of similarity-based measures.

future common neighbors, defining three classes of neighbors.

Recently, some studies have been reported on graph representation, and applied to a range of areas including link prediction in graphs (Radmanesh, Rezaei, Al Khafaf, & Jalili, 2020). These methods first embed the network connectivity into a low-dimensional space, where similarity between nodes can be obtained by comparing the embedded vectors. Then, link prediction is performed based on these similarities. DeepWalk (Perozzi, Al-Rfou, & Skiena, 2014) and node2vec (A. Grover & Leskovec, 2016) are two well-known graph embedding methods that have been applied to the link prediction problem. DeepWalk offers a fixed-length random walk, as opposed to the more flexible random walk provided by node2vec. A drawback of node2vec is parameter sensitivity, and the values selected for the number of walks, length of walk, and neighborhood size directly affect the performance of the method (A. Grover & Leskovec, 2016). LINE (Tang, et al., 2015) is based on graph embedding, and uses matrix factorization for the link prediction. Another class of methods based on graph embedding includes neighborhood aggregation and convolutional encoders. Graph Convolutional Networks (GCNs) (Kipf & Welling, 2016) is a popular method based on neighborhood aggregation and convolutional encoders. A common drawback of these methods is that they often require preprocessing, resulting in high computational complexity.

The challenges involved in local measures can be categorized broadly as follows.

- If the target pair of nodes for which the link prediction problem is considered does not include a common neighbor, many local measures based on common neighborhood will not predict any likelihood for link establishment. However, there are some instances in real systems where new relationships are created between nodes with no common neighborhood. Thus, this phenomenon cannot be described by local measures.
- The neighborhood directions are not considered. In many available local measures, neighborhood is defined in terms of proximity. In other words, most of these measures have not been defined particularly for directed graphs.

In this paper, new local similarity measures based on neighborhood direction are presented. The proposed measures consider the relationships between neighbors and the number of common neighbors. The contributions of the proposed link prediction method are as follows:

- Redefinition of conventional local measures in terms of neighborhood direction;
- Introducing a novel hybrid local measure using neighborhood direction and the *hub* and *authority* of common neighbors;
- Application in both supervised and unsupervised modes.

The rest of the paper is organized as follows. In Section 2, some existing local similarity measures are briefly reviewed. The motivation of the research and properties of the proposed approach are detailed in Section 3. In Section 4, the experimental results on some datasets are discussed. Finally, a summary of the research and suggestions for future works are included in Section 5, *Statistical analysis in section 6*.

2. Related works

The concept of link prediction in social networks was introduced by Liben-Nowell *et al.* (Liben-Nowell and Kleinberg, 2007). They investigated six similarity measures, which were based upon network topology. Al Hasan, Chaoji, Salem, and Zaki (2006) used similarity measures as classification features and investigated the efficiency of their model using different classifiers. Many more researchers have introduced different approaches to solve the link prediction problem. Generally, link prediction approaches can be categorized into three classes (Lü & Zhou, 2011):

- similarity-based algorithms
- maximum likelihood methods
- probabilistic models.

Similarity-based methods are based on calculations of the amounts of similarity within pairs of nodes. The term *proximity* is also used for expression of similarity. In these methods, a score s_{xy} is considered for each pair of nodes x and y. A score is considered for each potential link, and the higher that score, the more likely the establishment of the link between the nodes. In methods based on maximum likelihood estimations, a series of rules are extracted, and the probability of link establishment is then calculated. In probabilistic methods, an abstract model of the network is generated, based on which link prediction is made. Methods based on maximum likelihood estimations and probabilistic models often result in better accuracy than similarity-based methods, but at the price of high computational complexity (Lü & Zhou, 2011). The high computational complexity of these methods makes similarity measures more applicable in realistic scenarios for large-scale networks.

Wang, Xu, Wu, and Zhou (2015) presented a general framework for the link prediction problem. This framework includes two general types of method: similarity-based ones, which are unsupervised methods, and supervised machine learning methods. A wide range of the existing methods are based on some kind of similarity estimation within node pairs. Fig. 1 presents a classification of various similarity-based link prediction methods. In the following, we provide further details on the link prediction methods existing in the literature.

2.1. Local measures

Most local measures (excluding Preferential Attachment) function according to common neighborhood. Some well-known measures in this class are as follows.

- Common neighbors (CN) (Liben-Nowell and Kleinberg, 2007). This measure uses the intersection of the common neighbors of the two nodes for calculation of similarity. *Similarity s_{xy} between two nodes x and y is obtained as:*

$$CN_{(x,y)} = |\Gamma(x) \cap \Gamma(y)| \tag{1}$$

where $\Gamma(\boldsymbol{x})$ is the set of neighbors of node x. Although simple, this measure exhibits relatively good accuracy on most real-world networks (Martínez, et al., 2016).

- Jaccard coefficient (Jaccard, 1901). This similarity measure is defined as follows.

$$JC_{(x,y)} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|} \tag{2}$$

It is in fact a normalized version of CN.

- Adamic Adar (AA). This measure has been proposed originally for calculation of the similarity between two web pages (Adamic & Adar, 2003), and is defined as follows.

$$AA_{(x,y)} = \sum_{z \in \Gamma(X) \cap \Gamma(y)} \frac{1}{\log(\Gamma(z))} \tag{3}$$

In this measure, nodes with fewer neighbors are assigned higher scores.

- Resource allocation (RA) (Zhou, Lü, & Zhang, 2009). This measure has been adopted from the physical process of resource allocation. It is similar in form to AA. These two measures penalize common neighbors with high degrees, but RA is stricter in that regard than AA. The measure is calculated as follows.

$$RA_{(x,y)} = \sum_{z \in \Gamma(X) \cap \Gamma(y)} \frac{1}{\Gamma(z)} \tag{4}$$

AA and RA often have close prediction accuracy for networks with small average degrees; while RA exhibits better efficiency for networks with high average degrees (Feng, Zhao, & Xu, 2012).

The time complexity of RA is $o(vk^3)$, where v is the number of nodes, and k is the largest node degree on the graph. Sarkar, Chakrabarti, and Moore (2011) demonstrated that there are restrictions on use of the number of common neighbors for obtaining the similarity between two nodes. One of these restrictions is that neighborhood direction in the pair of nodes is disregarded.

- Preferential Attachment (PA) (Barabâsi, et al., 2002). Node degree distribution is of the power-law type in many real-world networks. This characteristic accounts for the existence of scale-free networks. The measure is defined as follows.

$$PA_{(x,y)} = |\Gamma_x| \times |\Gamma_y| \tag{5}$$

Unlike most local measures, this one is not based on common neighbors, and its time complexity is $o(vk^2)$.

- Resource Allocation Based on Neighbor interaction(RA-CNI)(Zhang, Zhang, Yang, & Yang, 2014)

This measure is based on the interactions between common neighbors with a focus on resource allocation, where each node sends a unit of information to its neighbors, the return of which is considered at the same time. The measure is represented as follows.

$$RZ - CNI_{(xy)} = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{|\Gamma_z|} + \sum_{ei,j \in Ei, |\Gamma_i| < |\Gamma_j|, i \in \Gamma_i, j \in \Gamma_j} \left(\frac{1}{|\Gamma_i|} - \frac{1}{|\Gamma_j|} \right) \tag{6}$$

- Sorensen (Sørensen, et al., 1948)

This measure was employed by Sørensen in 1948 for comparing the similarity between different data samples from ecological communities. Despite the similarity to the Jaccard coefficient, it is less sensitive to outliers. The measure is defined as follows.

$$S0_{(xy)} = \frac{2 \times |\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x)| + |\Gamma(y)|} \tag{7}$$

- Hub Promoted Index(HPI) (Ravasz, Somera, Mongru, Oltvai, & Barabási, 2002)

This measure was first proposed by Ravasz in 2002 for investigation of the modular structure of the metabolic network. A network of this type has a hierarchical structure with small internal modules, which are all separate from each other. The measure mainly aims at preventing links from being established merely between hubs and promoting establishment of links between nodes with low degrees and hubs, and it is defined as follows.

$$HPI_{(xy)} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\min\{|\Gamma(x)|, |\Gamma(y)|\}} \tag{8}$$

- Hub Depressed Index(HDI) (Ravasz, et al., 2002)

$$HDI_{(xy)} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\max\{|\Gamma(x)|, |\Gamma(y)|\}} \tag{9}$$

- Local Leicht-Holme-Newman Index (LLHN) (Leicht, Holme, & Newman, 2006)

This measure is the normalized variant of common neighbors, and is defined as follows:

$$LLHN_{(xy)} = \frac{|\Gamma_x \cap \Gamma_y|}{|\Gamma_x||\Gamma_y|} \tag{10}$$

- Salton Index (SA)(McGill, 1983)

This measure is known as the cosine measure, and is directly related to the Jaccard coefficient. It is defined as follows.

$$SA_{(xy)} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma_x| \times |\Gamma_y|}} \tag{11}$$

- Mutual Information(MI) (Tan, Xia, & Zhu, 2014)

This measure is calculated as follows using the information on the link itself and the common neighbors.

Table 1
An overview of local similarity measures. Normalized similarity score is value between zero and one.

Measure	normalized similarity score	Main Property	Description	Time complexity
Common Neighbors (CN)	No	Obtaining the number of common neighbors	Simple and intelligible	$O(VK^3)$
Adamic Adar (AA)	No	Common neighbors with fewer neighbors	Punishment of common neighbors with high degrees	$O(VK^3)$
Resource Allocation (RA)	No	Similar to AA	Common neighbors with higher degrees are punished more.	$O(VK^3)$
Resource Allocation Based on Common Neighbor Interactions (RA-CNI)	No	A hybrid of RA and CN	Relatively complex. For a pair of nodes like x, y, x should have fewer neighbors than y, which results in a bit greater computation than in CN and RA.	$O(VK^4)$
Jaccard (JC)	Yes	Ratio of the intersection to the union of the common neighbors	Simple and intelligible. Requires the union and intersection in computation.	$O(VK^3)$
Sørensen (SO)	Yes	Nodes with lower degrees are more likely to establish links.	Consideration of nodes with lower degrees	$O(VK^3)$
Preferential Attachment (PA)	No	It will be more likely to establish links between nodes with high degrees.	Not based on common neighbors, considering nodes with higher degrees	$O(VK^3)$
Hub Promoted (HPI)	Yes	Link probability is specified by lower node degrees.	Simple and intelligible	$O(VK^3)$
Hub Depressed (HDI)	Yes	Link probability is specified by higher node degrees.	Simple and intelligible	$O(VK^3)$
Salton Index (SA)	Yes	For calculation of the cosine similarity between two nodes	Cosine similarity	$O(VK^3)$
Local Leicht-Holme-Newman (LLHN)	Yes	Links are more likely to be established between couples of nodes with many common neighbors.	Pairs of nodes have more common neighbors than expected.	$O(VK^3)$
Mutual Information (MI)	Yes	Use of probability rules and common neighbors for obtaining the similarity between couples of nodes	Complex	$O(VK^6)$
CAR-Based Indices (CAR)	Yes	Use of a notion known as local community (LC) and redefinition of the AA, RA, and CN measures on that basis	Complex. There will be a link between two nodes if the common neighbors belong to one class, which is highly powerful in terms of internal cohesion.	$O(VK^4)$
Functional Similarity Weight (FSW)	Yes	Derived from the Sørensen measure	Complex, requiring a constant coefficient λ to be calculated	$O(VK^3)$
Individual Attraction Index (IA)	Yes	There will be a link between two nodes with common neighbors if there is a firm connection between their common neighbors.	Relatively simple, similar to RA	$O(VK^3)$
Local Naïve Bayes (LNB)	Yes	Each common neighbor will have a different role in terms of degree or influence in establishment of a link, so the probability of the link can be estimated by the theories of likelihood.	Complex, with high computation. f(z), which specifies node influence, involves additional computation	$O(vO(f(z))+vK^3)$
CNDP		Distinguishing between the common neighbors	Simple and intelligible	$O(VK^3)$

$$S_{xy} = -I(e_{x,y}|\Gamma_x \cap \Gamma_y) = -I(e_{x,y}) + \sum_{z\in\Gamma_x\cap\Gamma_y} I(e_{x,y};z) \tag{12}$$

$$I(e_{x,y}) = -log_2^{(1-\prod_{i=1}^{|\Gamma_x|}\frac{|E|-|\Gamma_x|-i+1}{|E|-i+1})} \tag{13}$$

$$I(e_{x,y};z) = \frac{1}{|\Gamma_z|(|\Gamma_z|-1)} \sum_{u,v\in\Gamma_z, u\neq v} (I(e_{u,v}) - I(e_{u,v}|z)) \tag{14}$$

$$I(e_{x,y}|z) = -log_2^{\frac{|\{e_{x,y}:x,y\in\Gamma_z, e_{x,y}\in E\}|}{2|\Gamma_z|(|\Gamma_z|-1)}} \tag{15}$$

- CAR-Based Indices (CAR) (Cannistraci, Alanis-Lobato, & Ravasi, 2013)

Based on this measure, the probability of link establishment between two nodes increases if their common neighbors are members of a group of powerful internal links known as local associations. This assumption allows us to obtain a larger number of important nodes with internal links to the other neighbors. The CAR-based version of common neighbors is defined as follows.

$$S_{xy} = \sum_{z\in\Gamma_x\cap\Gamma_y} 1 + \frac{|\Gamma_x \cap \Gamma_y \cap \Gamma_z|}{2} \tag{16}$$

- Function Similarity weight (FSW) (Chua, Sung, & Wong, 2006)

This measure was derived from the Sørensen index. On that basis, the probability that there is interaction from node x to node y on a directed graph is independent of the probability that there is interaction from y to x. The measure can be extended to undirected graphs, and is defined as follows.

$$S_{xy} = \left(\frac{2|\Gamma_x \cap \Gamma_y|}{|\Gamma_x - \Gamma_y| + 2|\Gamma_x \cap \Gamma_y| + \lambda}\right)^2 \tag{17}$$

- *individual* Attraction index (IA) (Dong, Ke, Wang, & Wu, 2011)

IA is defined as:

$$IA_{(xy)} = \sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{e_z}{k(z)} \tag{18}$$

where e_z is the number of edges between nodes x and y and their common neighbors, and $k(z)$ is the degree of node z. Standardized IA (SIA) is defined as follows:

$$SIA_{(xy)} = \sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{k(z)} \times \frac{e_z+2}{|\Gamma(x)\cap\Gamma(y)|} \tag{19}$$

- Local Naıve Bayes (LNB) (Z. Liu, Zhang, Lü, & Zhou, 2011)

This measure assumes that each of the common neighbors has a different role and influence in the network, which can be used for estimation of the probability of establishment of links between nodes in the

future. The measure is defined as:

$$LNB_{(xy)} = \sum_{z \in \Gamma_x \cap \Gamma_y} f(z)\log(oR_z) \tag{20}$$

where o is a constant value, calculated as follows.

$$o = \frac{P_{unconnected}}{P_{connected}} = \frac{\frac{1}{2}V(|V|-1)}{|E|} - 1 \tag{21}$$

The role or influence of each node is specified by R_z, which is calculated as follows.

$$R_z = \frac{2\left|\{e_{x,y} : x, y \in \Gamma_z, e_{x,y} \in E\}\right| + 1}{2\left|\{e_{x,y} : x, y \in \Gamma_z, e_{x,y} \notin E\}\right| + 1} \tag{22}$$

- CNDP: Common neighbors degree penalization (Rafiee, Salavati, & Abdollahpouri, 2020)

In this measure, degrees of the common neighbors are penalized by the values of c and β, the former indicating average clustering coefficient, and the latter being a constant coefficient. If the common neighbors of nodes x and y are friends, it is more likely that a link will be established between them. The measure is formulated as:

$$CNDP_{(x,y)} = \sum_{z \in \Gamma_x \cap \Gamma_y} |C_z|(|\Gamma_z|)^{-\beta c} \tag{23}$$

where Γ_z represents the number of common neighbors, and C_z indicates those with links between them.

A comparison of the local measures is provided in Table 1.

2.2. Global measures

This class of measures functions according to path and random walk. The shorter the distance between two nodes on a social network, the higher the chances of establishment of a link between them. The time complexity of most global measures is $o(v^2)$. Since the scores for all node pairs should be examined, the computational complexity of measures of this type can make them impractical for large-scale networks. Two important measures in this class are SimRank and Random Walks with Restart (RWR).

2.2.1. Random walks with restart (RWR)

(Kleinberg, 1999) presented an idea referred to as Hyperlink-Induced Topic Search (HITS) for identification of important webpages. The main idea in HITS is to use hyperlinks as votes. The rank score of a page depends on the number of input links and the rank scores of the pages of the neighbors that have provided links to that page. If there is a link on page u to page v, the author of page u has actually confirmed the importance of page v. According to this idea, node u is referred to as *hub*, and node v is known as *authority*. The hubs point to the authorities.

Authority←Hub

On that basis, a score in terms of the hubs and authorities can be calculated for each node i, as follows.

$$\begin{cases} h_i \leftarrow \sum_{i:(j,i)\in E} a_i \\ a_j \leftarrow \sum_{i:(i,j)\in E} h_i \end{cases} \tag{24}$$

RWR is a contraction for Random Walks with Restart (Tong, Faloutsos, & Pan, 2006), which is the manipulated HITS algorithm. The rank of a node on a graph is proportionate to the probability of reaching that node with random walks on the graph. Assume that you start moving from an arbitrary node x , exiting the node with a probability of α, and continue the random walk, returning to node x with a probability of $1 - \alpha$. The RWR measure is defined as:

$$\vec{r_i} = \alpha M \ \vec{r_i} + (1-\alpha)\vec{e_i} \tag{25}$$

where M is transition probability matrix, $\vec{r}_i$ is a $1 \times v$ vector, and $\vec{e}_i$ is a $1 \times v$ vector, the i^{th} element of which is 1. Since the measure is asynchronous, the similarity within node pairs is calculated as follows:

$$s_{xy} = \vec{r}_y^{\,x} + \vec{r}_x^{\,y} \tag{26}$$

Blondel, Gajardo, Heymans, Senellart, and Van Dooren (2004) presented a measure for obtaining the similarity between the nodes of two directed graphs according to the hubs and authorities. The measure can also be used for obtaining the similarity within a pair of nodes on a single graph. Let G_a and G_b be two directed graphs with n_a and n_b nodes. The n_a x n_b matrix is a similarity matrix, in which entry (i, j) specifies the similarity of node i of matrix G_a to node j of matrix G_b. In a special case, it can be assumed that $G_a = G_b = G$. The similarity measure is then defined as:

$$S(t) = \frac{AS(t-1)A^T + A^T S(t-1)A}{|AS(t-1)A^T + A^T S(t-1)A||_F} \tag{27}$$

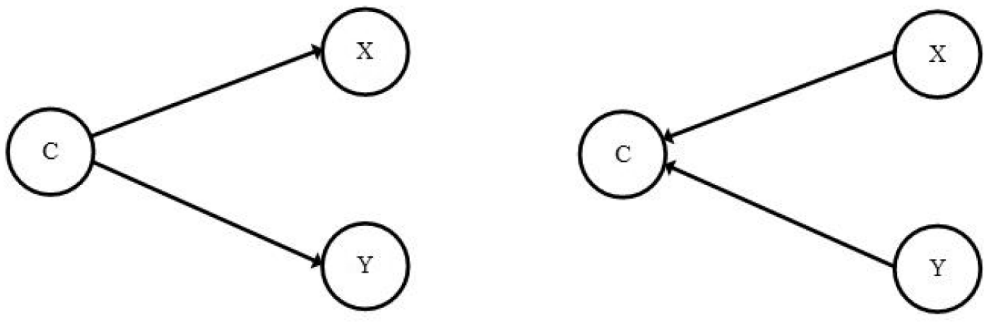
where A is the adjacency matrix of the graph, $S(0) = I$, and $||M||_F$ is the Frobenius normal form of the matrix. The above formula is a recursive equation, converging in the c^{th} iteration and having a time complexity of $O(v^2)$.

2.2.2. SimRank

Based on this measure (Jeh & Widom, 2002), two nodes are similar to each other if referred to by similar nodes. This measure is defined as:

$$Sim_{(xy)} = \frac{\beta}{|\Gamma(x)||\Gamma(y)|}\sum_{i\in\Gamma(x)}\sum_{j\in\Gamma(y)} s(i,j) \tag{28}$$

where $s(i, i) = 1$, and β is a damping factor between 0 and 1.



(a).Common neighbor IN (b).Common neighbor OUT

Fig. 2. Common neighbors IN and OUT.

2.3. Quasi-local measures

This class of measures provides a balance between local and global measures. With local measures, the priority is to keep computational complexity low, while accuracy is prioritized for global measures. Quasi-local measures seek to provide a balance between the two methods. Two well-known measures in this class include Local Random Walk (LRW) (W. Liu & Lü, 2010) and FriendLink (FL) (Papadimitriou, Symeonidis, & Manolopoulos, 2012)

2.4. Similarity measures based on in-degree and out-degree (neighborhood direction)

Most local measures for undirected graphs are based on adjacency. If a pair of nodes x and y is adjacent to a node a, this node is considered as their common neighbor. In-degree and out-degree are often disregarded in these measures. In similarity measures based on neighborhood direction, two types of neighborhood are defined (Laishram, 2015), on the basis of which the AA, RA, and CN measures are redefined. A node y is a *neighbor out* of a node x if there is an edge out of x to y. The set of neighbors out of x can be represented as $\Gamma_O^{(x)}$. A node y is a *neighbor in* of a node x if there is an edge out of y to x. The set of neighbors in of x can be indicated as $\Gamma_i^{(x)}$. Given these two definitions, the local measures are redefined below.

2.4.1. Common neighbor out (CNO)

In a directed graph, a node c is the common neighbor out of a pair of nodes x and y if there are two edges out of the pair to node c (Fig. 2.b). Based on this new definition, the common neighbor out measure is defined as:

$$CN_o^{(x,y)} = \left|\Gamma_o^{(x)} \cap \Gamma_o^{(y)}\right| \tag{29}$$

where $\Gamma_o^{(x)}$ is the set of neighbors out of node x.

2.4.1.1. Common neighbor in (CNI). In a directed graph, as in Fig. 2.a, a node c is the common neighbor in of nodes x and y if there are two edges out of node c to the pair. The common neighbor in measure is calculated as:

$$CN_i^{(x,y)} = \left|\Gamma_i^{(x)} \cap \Gamma_i^{(y)}\right| \tag{30}$$

where $\Gamma_i^{(x)}$ is the set of neighbors in of node x.

2.4.1.2. Common neighbor in out (CNIO). The combination of the common neighbors in and out is calculated as follows.

$$CN_{io}^{(x,y)} = S_i^{x,y} + S_o^{x,y} \tag{31}$$

Given the definitions of the common neighbors in and out, the AA and RA measures are redefined as follows.

Adamic Adar Out (AAO):

$$AA_o^{(x,y)} = \sum_{z_o \in \Gamma(x_o) \cap \Gamma(y_o)} \frac{1}{\log(\Gamma(z_o))} \tag{32}$$

Adamic Adar In (AAI):

$$AA_i^{(x,y)} = \sum_{z_i \in \Gamma(x_i) \cap \Gamma(y_i)} \frac{1}{\log(\Gamma(z_i))} \tag{33}$$

Adamic Adar In Out (AAIO):

$$AA_{io}^{(x,y)} = AA_I^{x,y} + AA_O^{x,y} \tag{34}$$

Resource Allocation Out (RAO):

$$RA_O^{(x,y)} = \sum_{z_O \in \Gamma_O^{(x)} \cap \Gamma_O^{(y)}} \frac{1}{\Gamma_o^{(z)}} \tag{35}$$

Resource Allocation In (RAI):

$$RA_i^{(x,y)} = \sum_{z_i \in \Gamma_i^{(x)} \cap \Gamma_i^{(y)}} \frac{1}{\Gamma_i^{(z)}} \tag{36}$$

Resource Allocation In Out (RAIO):

$$RA_{io}^{(x,y)} = RA_I^{(x,y)} + RA_O^{(x,y)} \tag{37}$$

Measures of neighborhood direction properly consider the direction of neighborhood in the link prediction process. However, suffer from the fact that two nodes without any common neighbors are unlikely to stablish a link in the future.

J. Li, et al. (2020) presented a link prediction measure based on reciprocal links, where $E^t = \{(u,v) \in E and (v,u) \in E\}$. Reciprocal coefficient is formulated as $p = |E^t|/|E|$, and a weight value is defined for each edge as:

$$w_{xy} = a_{xy} + p.\frac{a_{yx}}{k_y^{out}} \tag{38}$$

where a_{xy} is the value in row x and column y on the adjacency matrix, and k_y^{out} is the out-degree of node y. The conventional measures are redefined as in Equations (38) to (43).

$$IRW - DCN_{(x,y)} = \sum_{z \in V} w_{xz}.w_{zy} \tag{39}$$

$$\boldsymbol{IRW - DAA_{(x,y)} = \sum_{z \in V} \frac{w_{xz}.w_{zy}}{log(1+s_z)}} \tag{40}$$

$$\boldsymbol{IRW - DRA_{(x,y)} = \sum_{z \in V} \frac{w_{xz}.w_{zy}}{S_Z}} \tag{41}$$

$$\boldsymbol{IRW - Bifan_{(x,y)} = \sum_{z \in V} w_{xz}.w_{z'z}.w_{z'y}} \tag{42}$$

$$\boldsymbol{DRW_{(x,y)} = IRW_{(x,y)} + \lambda' IRW_{(yx)}} \tag{43}$$

$$\boldsymbol{DRW_{(x,y)} = IRW_{(x,y)} + p.\frac{IRW_{(y,x)}}{k_y^{out}}} \tag{44}$$

2.5. Similarity measures based on network motifs

It has been demonstrated that many real-world networks are constructed from structural blocks referred to as network motifs. Motifs are repetitive patterns of connectivity between network nodes, which are significantly more frequent in the network than in its randomized counterpart (Aghabozorgi & Khayyambashi, 2018). Schall (2014) presented a method of link prediction for directed graphs using network motifs. Aghabozorgi and Khayyambashi (2018) presented a local measure using network motifs as:

$$\text{Triadic}_{(x,y)} = \frac{\sum_{z \in \Gamma(x) \cap \Gamma(y)} \Phi(x,y,z) \times 1/13}{|\Gamma(x) \cap \Gamma(y)|} \tag{45}$$

where $\Phi(x,y,z)$ is the number of motifs that nodes x and y form together with their common neighbor z. The in and out directions of the common neighbor have been disregarded in this measure, which is based only on the structure of the motif.

Y. Liu, Li, and Xu (2019) introduced two link prediction methods based on Naïve Bayes model. They investigated the different effects of neighboring edges and nodes in link prediction.

2.6. Representation learning on graphs

Novel approaches have been recently proposed on graph-based machine learning, known as Representation Learning on Graphs. A major challenge in these approaches is to properly represent the graph nodes. Embedding provides a solution to the problem, where the connectivity matrix is mapped onto a low-dimensional vector space.

Node embedding

This approach encodes every graph node as a low-dimensional vector. The distance between the vectors in the d-dimensional space represents the similarity between the nodes in the original graph. The node embedding approach includes two major functions known as the encoder and the decoder (Hamilton, Ying, & Leskovec, 2017).

Encoder. It is a function that maps each graph node onto a low-dimensional vector, and is defined as

$$ENC : V \rightarrow \mathbb{R}^d \tag{46}$$

where V is the number of graph nodes, $Z_i \in \mathbb{R}^d$ is the embedding vector for node $v_i \in V$, and d specifies the dimensions of vector Z_i.

Decoder. It is a function defined as follows:

$$DEC : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+ \tag{47}$$

The decoder maps embedding of each pair of nodes into a value, *i.e.* the similarity measure of the pair of nodes. The similarity measure is defined as:

$$DEC\big(ENC(v_i), ENC(v_j)\big) = DEC(Z_i, Z_j) \approx S\mathcal{G}(u_i, v_j) \tag{48}$$

where $S\mathcal{G}(v_i, v_j)$ is a similarity measure defined by the user based on the graph. The aim is to optimize the encoder and decoder functions, where the latter specifies the similarity between the two nodes. For this purpose, the loss function is defined as:

$$\mathcal{L} = \sum_{(u_i,v_j)\in D} \ell\big(DEC(Z_i, Z_j), S\mathcal{G}(u_i, v_j)\big) \tag{49}$$

where $\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ and is defined by the user. It specifies the difference between $DEC(Z_i, Z_j)$ and $S\mathcal{G}(u_i, v_j)$. D represents the set of training node pairs.

Recently, a number of efficient techniques have been proposed for graph embedding, such as Deep Walk (Perozzi, et al., 2014) and node2vec (A. Grover & Leskovec, 2016). In these methods, *DEC*, *ENC*, and the loss functions are defined as follows:

$$ENC(v_i) = Z_{V_i} \tag{50}$$

where $Z \in \mathbb{R}^{d\times|v|}$ is a matrix containing embedding vectors for all the nodes, and $V_i \in \mathbb{I}^{|v|}$ is a vector on which all the values are zero except those on the V_i node column.

$$DEC(Z_i, Z_j) = \frac{\exp(Z_i^{\mathrm{T}}.Z_j)}{\sum_{v_k\in V} exp(Z_i^T.Z_k)} \approx p_{\mathcal{G},t}(v_i|v_j) \tag{51}$$

where $p_{\mathcal{G},t}(v_i|v_j)$ is the probability that node v_i meets node v_j on a random walk of length t.

$$\mathcal{L} = \sum_{(u_i,v_j)\in D} -log(DEC(Z_i, Z_j)) \tag{52}$$

where D is a training set, a sampling of the set of nodes obtained in random walks beginning from an arbitrary node.

The main difference between DeepWalk and node2vec lies in how they formulate Eq. (45) (Hamilton, et al., 2017). DeepWalk uses the hierarchical softmax technique for calculation of the normalizing factor, and increases calculation speed using the binary tree structure. Node2-vec, on the other hand, approximates the normalizing factor using random negative samples.

Large-scale Information Network Embedding (LINE) is another method for graph embedding, which is not based on random walk (Tang, et al., 2015). In this method, shallow embedding is used for defining ENC, and DEC, as:

$$DEC(Z_i, Z_j) = \frac{1}{1 + exp(-Z_i^T.Z_j)} \approx p_{\mathcal{G}}(v_i, v_j) \tag{53}$$

where $p_{\mathcal{G}}(v_i, v_j) = A_{ij}$, and A_{ij} is the similarity adjacency matrix.

Topological Deep Network Embedding (TDNE) (Radmanesh, et al., 2020)

This is a semi-supervised deep learning embedding method. Since most real-world social networks are sparse, inputting the adjacency matrix to an auto-encoder highlights the zero elements. For the problem to be resolved, the non-zero elements are penalized more than the zero elements, and three loss functions are defined on that basis.

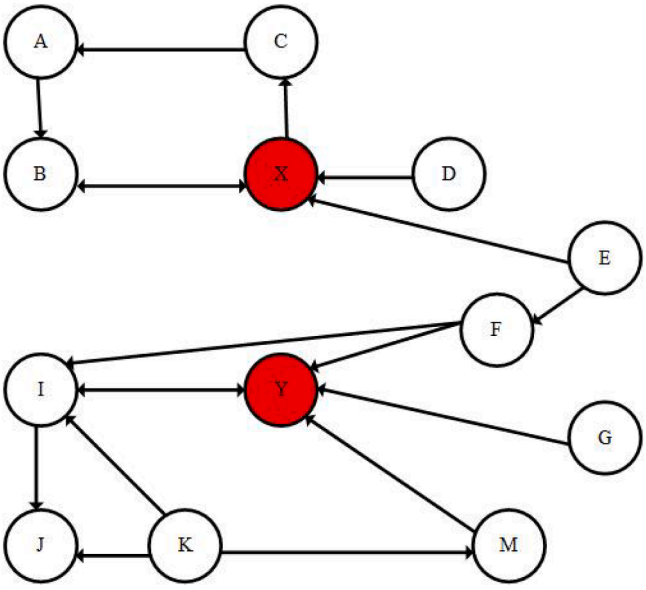


Fig. 3. Example of directed networks.

3. Motivation and proposed link prediction approach

3.1. Motivation

Most local measures are based on common neighbors. A problem with these methods is that if two nodes do not have a common neighbor, the probability of their friendship in the future will be assumed to be zero. However, there are some instances in real systems where nodes with no common neighbors have established friendships. Such local measures cannot describe this phenomenon, and one has to use richer information on the connectivity. For an illustrative example, let us consider nodes X and Y in Fig. 3. Their in- and out-degrees are higher than those of the other nodes. The two nodes are topologically similar to each other, and thus likely establish a relationship.

From the local measures perspective, however, the similarity of X and Y is assumed to be zero as they do not have a common neighbor.

$$AA_o^{(x,y)} = 0 RA_{io}^{(x,y)} = 0$$

$$AA_i^{(x,y)} = 0 CN_o^{(x,y)} = 0$$

$$AA_{io}^{(x,y)} = 0 CN_i^{(x,y)} = 0$$

$$RA_{\boldsymbol{o}}^{(\boldsymbol{x},\boldsymbol{y})} = 0 CN_{io}^{(x,y)} = 0$$

$$RA_{\boldsymbol{i}}^{(\boldsymbol{x},\boldsymbol{y})} = 0$$

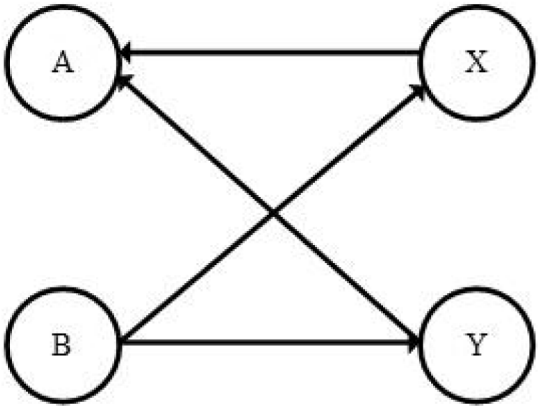


Fig. 4. An example of directed networks.

3.2. Proposed measure

The method proposed in this paper introduces a novel measure by combining the features obtained based on common neighbors and the hubs and authorities of the nodes. The proposed method overcomes the limitations of common neighbor based measures, while maintaining their simplicity. In the proposed measure, the hub, authority, and direction of the connection (in-out-neighbor) are used along with the information on the common neighbors. Let us consider Fig. 4, where node X has one neighbor in and one neighbor out, and so does node Y.

The common neighbor out of the pair of nodes X and Y is node A. Given the definitions of *hub* and *authority*, X and Y play the role of hubs for node A, which in turn functions as an authority for them. There are two edges out of node B to nodes X and Y. Therefore, node B plays the role of a hub for these nodes, and they are authorities for node B. The proposed method considers not only the information on the common neighbors but also information on the direction of the connectivity through their hub-ness and authority-ness. Here are three similarity measures based on this information.

3.2.1. Common neighbors hub authority (CN-HA)

The similarity between two nodes x and y based on both common neighbors and information on hub-ness and authority-ness is defined as:

$$CN-HA_{(xy)} = \left(\sum_{z\in\Gamma_o^{(x)}\cap\Gamma_o^{(y)}} Auth(z)\right) + (Hub(x)+Hub(y)) \tag{54}$$

where $Auth(z)$ is the authority of the set of common neighbors, and $Hub(x)$ indicates the hub-ness of node x.

Eq. (47) is composed of two parts. The first part involves $\sum_{z\in\Gamma_o^{(x)}\cap\Gamma_o^{(y)}} Auth(z)$, where z is the common neighbor out of the pair of nodes x and y, referring to z at the same time. Thus, z functions as *Auth* for x and y. This part considers neighborhood direction in the prediction process. The second part of the measure aims to resolve the problem of lack of common neighbors. If two nodes do not have a common neighbor, their similarity will be equal to the sum of their hubs, which might be non-zero depending on the structure of the network.

3.2.2. Common neighbors authority hub (CN-AH)

This measure is defined as follows.

$$CN-AH_{(xy)} = \left(\sum_{z\in\Gamma_i^{(x)}\cap\Gamma_i^{(y)}} Hub(z)\right) + (Auth(x)+Auth(y)) \tag{55}$$

3.2.3. Sum of common neighbors with hub and authority (SCNHA)

This measure is a combination of the above two similarity metrics, defined as follows

$$\mathrm{SCNHA}_{(xy)} = CN_HA_{(xy)} + CN_AH_{(xy)} \tag{56}$$

On that basis, the similarity score of the pair of nodes x and y for the graph in Fig. 4 is calculated as follows.

$$SCNHA_{(xy)} = 0.4066$$

$$CN-HA_{(xy)} = 1.15\times10^{-14}$$

$$CN-AH_{(xy)} = 0.4066$$

The proposed measures resolve the limitations of earlier methods by considering neighborhood direction and considering possibility of link creation between two nodes without common neighbors. Our experiments show that this simple modification significantly improves the performance of link prediction.

3.3. Time complexity of the proposed method

The time complexity of the proposed measures is composed of the time complexity of calculating *Hub* and *Auth* and the time required for obtaining the similarity between the nodes. First, the *Hub* and *Auth* values for each node are calculated and stored in two vectors. The HITS algorithm is used for calculation of *Hub* and *Auth*. The complexity of the algorithm depends on three factors (N. Grover & Wason, 2012):

- number of iterations (i)
- number of nodes
- number of edges out of each node (o_i).

Therefore, the time complexity of the algorithm is as follows.

$$v + iv\sum_{i=1}^{v} o_i + v = v\left(2 + i\sum_{i=1}^{v} o_i\right) \tag{57}$$

Since the numbers of iterations and edges out of each node are much smaller than the number of nodes, time complexity can be simplified as $O(v)$.

The complexity of the CN-HA algorithm depends on two factors:

- intersection of the sets of neighbors out of nodes x and y
- calculation of $Auth(z)$, where z is the set of neighbors out of nodes x and y.

The number of common neighbors out of a pair of nodes is directly related to the out-degree of the pair. Let k be the maximum out-degree of nodes x and y. Each neighboring node of x and y contains a set of k members. The time complexity of the intersection of the two sets is $O(k)$ based on the hash table. Therefore, the intersection is calculated for all the nodes at $O(vk^2)$. The complexity of the measure is calculated as

$$O(g(n)) + O(vk^2 f(n)) \tag{58}$$

where $g(n)$ is the time complexity of *Auth*, which is $O(v)$, and $f(n)$ is the time required for obtaining the similarity between the nodes. Since $Auth(z)$ has been calculated in the previous step, the time complexity is $O(1)$. Since the maximum number of neighbors out is k, $f(n) = k$. Therefore, the complexity of the measure is as follows.

$$O(v) + O(vk^3) = O(vk^3) \tag{59}$$

The complexity of CN-AH is$O(vk^3)$, where k is the maximum in-degree of nodes x and y. The time complexity of the SCNHA measure is $O(2vk^3)$, since the number of neighbors in and out is calculated. Based on Table 1, the time complexity of the proposed measure is comparable to that of local measures.

Table 2
Networks under investigation.

Name	Type	Nodes	Edges	Description
Political Blogs	Directed	1222	19,021	Network of hyperlinks between weblogs on US politics
Kohonen	Directed	4470	12,731	Articles with the topic *self-organizing maps* or references to *Kohonen, T*
SmaGri	Directed	1059	4922	Citations to Small & Griffith and descendants
Wiki-Vote	Directed	7115	103,689	Wikipedia who-votes-on-whom network

4. Experiments and results

The measures proposed in this paper can be used for solving the link prediction problem in both supervised and unsupervised modes. For implementation of the methods, the Python programming language and the NetworkX, Pandas, node2vec, and NumPy libraries are used.

4.1. Datasets

The link prediction methods are applied on four directed networks, which are accessible at http://linkprediction.org/index.php/link/resource/data and https://snap.stanford.edu/data/wiki-Vote.html. Their characteristics are listed in Table 2.

4.2. Baseline methods for comparison

The measures proposed in this paper are compared to the following baseline methods:

- CN-IN; common neighbor in (Laishram, 2015)
- CN-OUT; common neighbor out
- AA-IN;Adamic Adar measure based on common neighbor in
- AA-OUT; Adamic Adar measure based on common neighbor out
- RA-IN; resource allocation measure based on common neighbor in
- RA-OUT; resource allocation measure based on common neighbor out
- Triadic; a measure based on triadic motifs (Aghabozorgi & Khayyambashi, 2018)
- Node2vec (N2V); we consider the parameters of the model as d = 100, number walk = 18, and length walk = 100 (A. Grover & Leskovec, 2016).

4.3. Unsupervised link prediction

In an unsupervised approach, a score is calculated for each unconnected pair of nodes based on the similarity measure. The higher is the score, the greater would be the similarity between the two nodes and the probability that a link is established between them in the future. A descending list of scores is then developed, and links at the top of the list are those that are predicted to occur more probably in the future.

To examine the prediction accuracy of different similarity measures in unsupervised mode, let us denote the available edges by E, referred to as the *observed links*. These edges are then divided randomly into two parts, test and training, indicated by E^T and E^P, respectively.

It is given that $E = E^P \cup E^T$ and $E^T \cap E^P = \emptyset$. Two standard measures, namely AUC (Hanley & McNeil, 1982) and Precision (Herlocker, Konstan, Terveen, & Riedl, 2004), are used to assess the performance of the link prediction methods. To obtain AUC, a number of links are first selected randomly from E^t; these are referred to as *misslink*s. Then, some links are selected randomly from the set U/E, where U is the Universal set; these are called *nonexistentLinks*. AUC is interpreted as the probability that the *misslink* scores, calculated for these links through the link prediction method, are greater than the *nonexistentLink* scores (Lü & Zhou, 2011). AUC is obtained as:

$$AUC = \frac{n^{'} + 0.5n^{''}}{n} \tag{60}$$

where n' is the number of times that the *misslink* scores are greater than the *nonexistentLink* scores, and n'' is the number of times that the two scores are equal. n is the total number of comparisons. If the scores are obtained from an independent distribution, the value of AUC is expected to be 0.5. Therefore, a value higher than 0.5 indicates better performance than in a purely random case. To obtain prediction accuracy, Precision is calculated for the *nonexistentLink* scores, and the obtained values are sorted in descending order. Then, L links with the highest scores are selected, and l is obtained as the number of those that have been predicted correctly. Precision is formulated as follows.

$$Precision = \frac{l}{L} \tag{61}$$

For calculation of the evaluation metrics, at least two snapshots from the networks are required. To this end, 10% of the links are eliminated randomly from the input graph $G = (V,E)$; the eliminated links are listed in *misslink*, and the remaining graph is denoted as $G_t = (V_t, E_t)$. The original graph, G, is considered as $G_{t+1} = (V_{t+1}, E_{t+1})$. The subscript t indicates time; *i.e.*, G_t indicates graph G at time t. Given that the networks used in the experiments are sparse, and the number of observed links is far smaller than *nonexistentLink*, it would be very time-consuming to calculate AUC. For comparison of the *misslink* and the *nonexistentLink* scores, therefore, 50% of the nonexistent links are selected randomly, and the AUC value is obtained as the average for ten independent iterations of the procedure.

The values in boldface indicate the best-performer on each network. As can be seen in Table 3, CN-AH shows the highest AUC value for Political Blogs, CN-HA

is the top-performer in terms of AUC for SmaGri and Kohonen, and SCNHA indicates the highest AUC value for Wiki-Vote. The CN-IN-OUT

Table 3
Mean and standard (Std.) deviation of AUC for different measures across four networks. For each network, the bold font shows the top-performer measure.

Data set		N2V	SCNHA	CN-HA	CN-AH	AA-IN-Out	AA-IN	AA-OUT	CN_I_O	CN-O	CN-I	RA-In-Out	RA-In	RA-Out	Triadic	P_Value
SmaGri	Mean	0.58	0.78	**0.8**	0.68	0.57	0.57	0.54	0.74	0.64	0.66	0.59	0.59	0.56	0.73	0
	Std. Deviation	0.04	0.02	0.01	0.02	0.02	0.02	0.01	0.01	0.01	0.02	0.04	0.01	0.03	0.02	
Wiki-Vote	Mean	0.76	**0.9**	0.89	0.89	0.53	0.56	0.58	0.84	0.76	0.75	0.53	0.52	0.55	0.82	0
	Std. Deviation	0.02	0.01	0.01	0.02	0.02	0.03	0.03	0.02	0.02	0.02	0.01	0.01	0.02	0.01	
Political Blogs	Mean	0.73	0.88	0.83	**0.9**	0.63	0.68	0.67	0.88	0.83	0.83	0.66	0.72	0.71	0.82	0
	Std. Deviation	0.02	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.01	0.02	0.02	0.01	0.01	0.01	
Kohonen	Mean	0.6	0.64	**0.8**	0.58	0.56	0.55	0.56	0.6	0.57	0.59	0.55	0.54	0.53	0.61	0
	Std. Deviation	0.03	0.02	0.02	0.06	0.06	0.06	0.08	0.05	0.05	0.05	0.03	0.02	0.02	0.03	

Table 4
Mean and standard (Std.) deviation of Precision for different measures across four networks. For each network, the bold font shows the top-performer measure. (L = 1 TO 200).

Data set		N2V	SCNHA	CN-HA	CN-AH	AA-IN-Out	AA-IN	AA-OUT	CN_I_O	CN-O	CN-I	RA-In-Out	RA-In	RA-Out	Triadic	P_Value
SmaGri	Mean	0.12	0.37	**0.5**	0	0.02	0	0.03	0.04	0.05	0.04	0.02	0	0.03	0	0
	Std. Deviation	0.01	0.02	0.02	0	0.01	0	0.01	0.02	0.02	0.02	0	0	0.01	0	
Wiki-Vote	Mean	**0.8**	0.73	0.65	0.79	0.14	0.08	0.18	0.76	0.69	0.79	0.14	0.08	0.18	0.11	0
	Std. Deviation	0.09	0.05	0.06	0.57	0.71	0	0	0	0.67	0.61	0.67	0	0	0	
Political Blogs	Mean	0	**0.3**	0.13	0.16	0.03	0.05	0.01	0.28	0.15	0.18	0.03	0.05	0.01	0.1	0.0001
	Std. Deviation	0	0.11	0.05	0.05	0.01	0.02	0.01	0.08	0.05	0.08	0.01	0.02	0.01	0.11	
Kohonen	Mean	0.18	0.39	0.65	0.07	0.51	0.41	**0.8**	0.73	0.72	0.64	0.51	0.41	0.77	0.15	0.0003
	Std. Deviation	0.15	0.15	0.06	0.03	0.14	0.17	0.14	0.05	0.05	0.15	0.14	0.17	0.14	0.15	

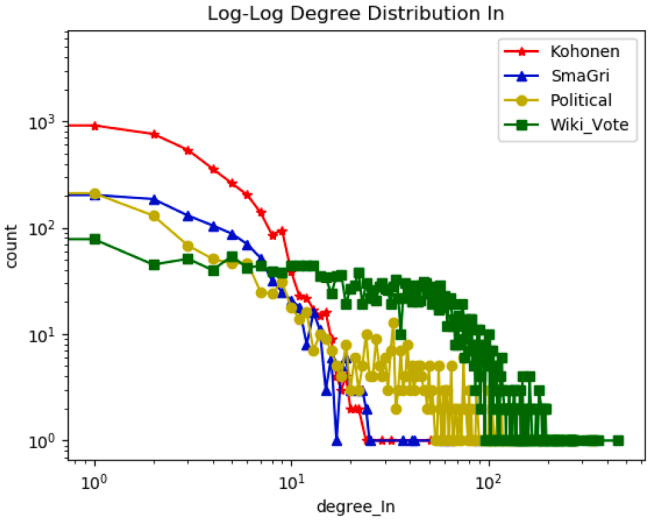


Fig. 5. Distribution of in-degree on the networks.

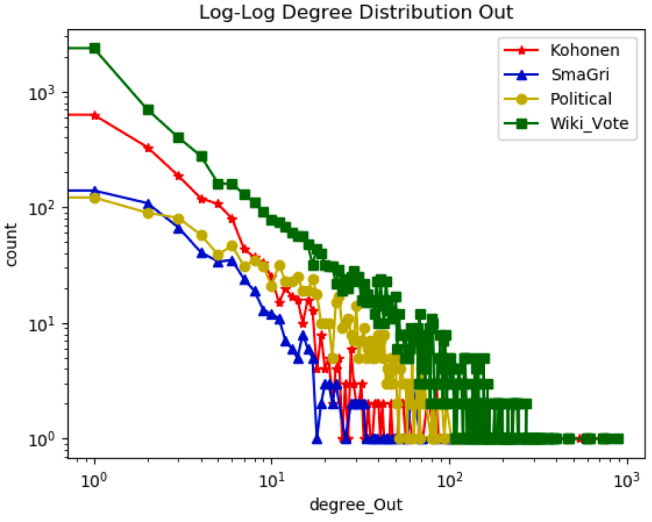


Fig. 6. Distribution of out-degree on the networks.

measure exhibits the highest Precision value for Wiki-Vote, while SCNHA has the best performance in terms of Precision for Political Blogs, and RA-OUT shows the highest Precision value for Kohonen, and CN-HA shows the highest Precision value for SmaGri and N2V shows the highest Precision value for Wiki-Vote (Table 4). All the top-performers are among the measures proposed here. As the value of *n'* increases, so does AUC. Nodes with high out-degrees increase the value of *n'* in CN-HA, ones with high in-degrees in CN-AH, and ones with high in- and out-degrees in SCNHA.

Figs. 5 and 6 show the in- and out-degree distributions for the networks. On the Wiki-Vote network, there are a large number of nodes with high in- and out-degrees, and the highest value of AUC concerns SCNHA. On the Kohonen network, there are nodes with high out-degrees, and the highest AUC value concerns CN-HA. On the Political Blogs network, there are nodes with high in-degrees, and the highest AUC value concerns CN-AH. These findings demonstrate the effect of connectivity direction on the accuracy of link prediction through the proposed measures.

4.4. *Supervised link prediction*

In a supervised approach to the link prediction, a binary classification is considered (Al Hasan, et al., 2006). Each non-adjacent node pair is regarded as a sample with a negative class label. Consider graph $G = (V,E)$, where $e \in E$, and $t(e)$ represents a link at time t. For $t < t^{'}$, $G[t, t^{'}]$ can be assumed to be a subgraph of G in the range between t and $t^{'}$. In a supervised approach, the learning period can be selected as $[t_0, t_0{'}]$ and the training period as $[t_1, t_1{'}]$ on the condition that $t_0^{'} < t_1$. The output is a list of links that are not there in range $G[t_0, t_0^{'}]$, but are predicted to be there in $G[t_1, t_1^{'}]$.

Supervised link prediction consists of two steps. The first step involves the provision of the information required for supervised learning and its collection in the form of a big table. The information includes the calculated similarity measures and inherent features of the nodes. The second step involves the use of a binary classification model based on the current instant image of the network to be applied for link prediction in the future. The data is divided into two parts: learning and test. There are two main challenges in the supervised mode. One is class imbalance (Lichtenwalter, Lussier, & Chawla, 2010), suggesting that the number of negative instances is far greater than that positive ones since most social networks are sparse. Link sampling is a method of resolving this issue (Han, Pei, & Kamber, 2011). The other issue concerns the use of an appropriate binary classifier with high efficiency. Linear discriminant analysis is appropriate for binary classifiers (Izenman, 2013)

(Pecli, et al., 2018) presented a supervised method of link prediction. They demonstrated that their feature selection method improves the efficiency of classifiers. Aghabozorgi *et al.* (2018) presented a supervised link prediction method. They introduced a similarity measure based on motifs of three, and examined the accuracy of their model using binary classification. Here, we use some well-known classifies, including LogisticRegression (Menard, 2002), GradientBoosting (J. H. Friedman, 2002), LinearDiscriminantAnalysis (Izenman, 2013), RandomForest (Breiman, 2001), and DecisionTree (Breiman, Friedman, Olshen, &

Table 5
Statistical significance *t*-test results for the AUC for Political Blogs ($k = 1$ To 10).

		N2V	SCNHA	CN-H-A	CN-A-H	AA-IN-Out	AA-IN	AA-OUT	CN_I_O	CN-O	CN-I	RA-In-Out	RA-In	RA-Out	Triadic	P-Value
LogisticRegression	Mean	0.79	0.79	0.78	0.79	0.77	0.78	0.77	0.8	0.8	0.79	0.78	0.77	0.78	**0.91**	0
	Std. Deviation	0.01	0.01	0	0.01	0.01	0.01	0.01	0	0.01	0.01	0.01	0.01	0.01	0.01	
GradientBoosting	Mean	0.9	0.92	0.9	0.92	0.88	0.89	0.88	0.93	0.91	0.91	0.88	0.89	0.88	**0.98**	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0	
LinearDiscriminant	Mean	0.88	0.86	0.86	0.85	0.85	0.85	0.85	0.88	0.85	0.87	0.86	0.85	0.86	**0.95**	0.0006
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0	
RandomForest	Mean	0.88	0.89	0.87	0.89	0.85	0.87	0.85	0.91	0.89	0.9	0.86	0.88	0.86	**0.96**	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0	
DecisionTree	Mean	0.79	0.79	0.78	0.79	0.77	0.78	0.77	0.8	0.79	0.8	0.78	0.78	0.77	**0.91**	0
	Std. Deviation	0.01	0.01	0	0.01	0.01	0.01	0.01	0	0.01	0.01	0.01	0.01	0.01	0.01	

Table 6
Statistical significance *t*-test results for the AUC for Kohonen (k = 1 to 10).

		N2V	SCNHA	CN-HA	CN-AH	AA-IN-Out	AA-IN	AA-OUT	CN_I_O	CN-O	CN-I	RA-In-Out	RA-In	RA-Out	Triadic	P_Value
LogisticRegression	Mean	**0.86**	0.81	0.81	0.8	0.8	0.8	0.8	0.84	0.82	0.82	0.81	0.81	0.8	0.84	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	
GradientBoosting	Mean	0.88	0.81	**0.9**	0.8	0.8	0.8	0.8	0.84	0.82	0.82	0.81	0.81	0.8	0.84	0.007
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	
LinearDiscriminant	Mean	**0.86**	0.81	0.81	0.8	0.81	0.81	0.8	0.83	0.81	0.82	0.81	0.81	0.8	0.85	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	
RandomForest	Mean	0.87	0.82	**0.88**	0.83	0.83	0.83	0.83	0.86	0.84	0.84	0.84	0.84	0.83	0.87	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	
DecisionTree	Mean	0.79	0.75	**0.8**	0.76	0.74	0.75	0.74	0.76	0.74	0.75	0.75	0.75	0.74	0.78	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.02	0.02	0.01	0.01	0.01	0.01	0.02	0.02	0.01	0.01	

Table 7
Statistical significance *t*-test results for the AUC for SmaGri ($k = 1$ to 10).

		N2V	SCNHA	CN-HA	CN-AH	AA-IN-Out	AA-IN	AA-OUT	CN-IN-OUT	CN-O	CN-I	RA-In-Out	RA-IN	RA-OUT	Triadic	P_Value
LogisticRegression	Mean	0.79	0.84	0.85	0.8	0.78	0.78	0.77	0.87	0.82	0.84	0.79	0.79	0.78	**0.89**	0
	Std. Deviation	0.02	0.01	0.01	0.01	0.02	0.02	0.02	0.01	0.01	0.01	0.02	0.02	0.02	0.01	
GradientBoosting	Mean	0.88	0.86	0.89	0.84	0.82	0.82	0.81	0.88	0.84	0.86	0.83	0.84	0.81	**0.93**	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.02	0.02	0.02	0.01	0.01	0.01	0.02	0.02	0.02	0.01	
LinearDiscriminant	Mean	0.79	0.82	0.78	0.8	0.78	0.78	0.77	0.8	0.77	0.82	0.79	0.79	0.77	**0.89**	0
	Std. Deviation	0.02	0.01	0.02	0.01	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.01	
RandomForest	Mean	0.85	0.85	0.87	0.82	0.81	0.82	0.8	0.87	0.83	0.84	0.8	0.81	0.79	**0.92**	0
	Std. Deviation	0.01	0.01	0.01	0.02	0.02	0.02	0.02	0.01	0.01	0.01	0.02	0.02	0.02	0.01	
DecisionTree	Mean	0.77	0.75	0.79	0.76	0.75	0.75	0.75	0.77	0.75	0.76	0.76	0.76	0.75	**0.84**	0
	Std. Deviation	0.01	0.01	0.01	0.01	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.02	

Stone, 1984). Negative sampling is used in this paper to resolve the problem of class imbalance.

$$numberofnegetiveinstance = \frac{CE}{CE + NE} \times NE \tag{62}$$

$$numberofposetiveinstance = \frac{NE}{CE + NE} \times NE \tag{63}$$

where *CE* represents the total number of graph edges, and *NE* indicates the number of unconnected ones.

For evaluation of these classifiers, the receiver operating characteristic (ROC) curve is used. To this end, the k-fold cross-validation method (k = 1 To 10) is used to generate training and test sets. Moreover, the area under ROC (AUC) is considered as a powerful measure for evaluation in cases of class disequilibrium (Hanley & McNeil, 1982). (AUC in

Table 8
Statistical significance *t*-test results for the AUC for Wiki-Vote ($k = 1$ to 10).

		N2V	SCNHA	CN-HA	CN-AH	AA-IN-Out	AA-IN	AA-OUT	CN_I_O	CN-O	CN-I	RA-In-Out	RA-In	RA-Out	MotifsTriad	P_Value
LogisticRegression	Mean	0.78	**0.91**	0.85	0.88	0.64	0.63	0.65	0.87	0.79	0.78	0.63	0.63	0.63	0.88	0
	Std. Deviation	0	0	0	0	0	0.01	0	0	0.01	0.01	0.01	0.01	0.01	0	
GradientBoosting	Mean	0.78	**0.91**	0.85	0.88	0.64	0.63	0.65	0.87	0.79	0.78	0.63	0.63	0.63	0.88	0
	Std. Deviation	0	0	0	0	0	0.01	0	0	0.01	0.01	0.01	0.01	0.01	0	
LinearDiscriminant	Mean	0.78	0.79	0.71	0.8	0.65	0.64	0.66	0.78	0.7	0.77	0.64	0.63	0.65	**0.88**	0
	Std. Deviation	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
RandomForest	Mean	0.82	0.9	0.88	0.88	0.7	0.69	0.71	0.87	0.81	0.82	0.7	0.69	0.71	**0.91**	0
	Std. Deviation	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
DecisionTree	Mean	0.79	0.84	0.8	**0.86**	0.74	0.74	0.74	0.8	0.77	0.78	0.74	0.74	0.74	0.85	0
	Std. Deviation	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

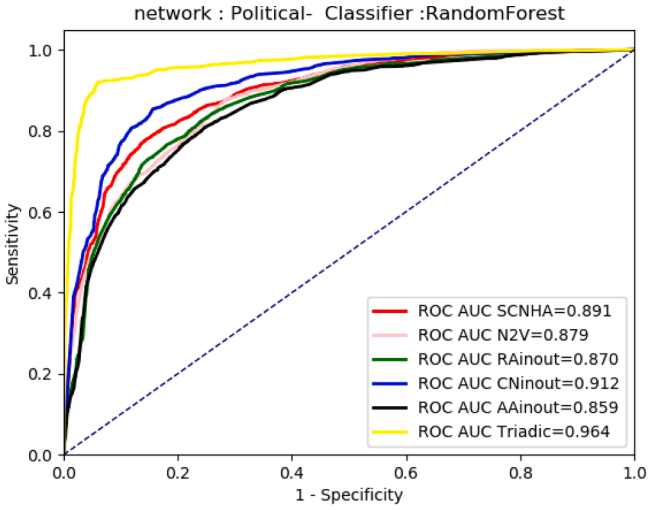


Fig. 7. ROC diagram for the Random Forest classifier over the Political Blogs network.

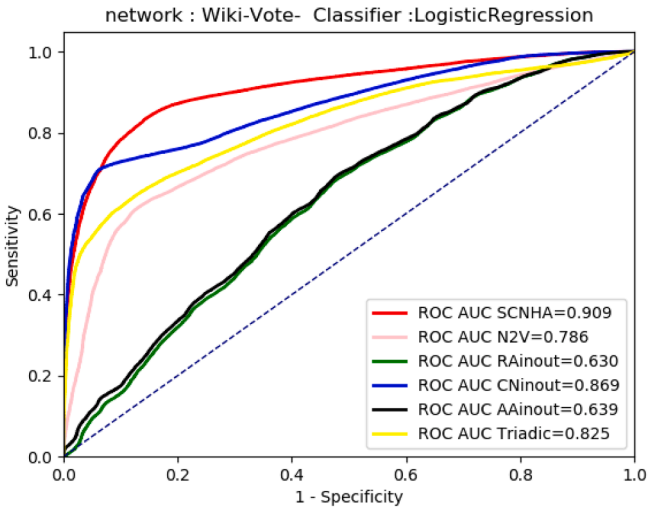


Fig. 8. ROC diagram for the LogisticRegression classifier over the Wiki-Vote network.

supervised mode is different from AUC in unsupervised mode). The mean AUC results obtained from the predictions appear in Tables 5 to 8.

Fig. 7 shows the ROC curves for the different measures using the RandomForest classifier over the Political Blogs network. It can be observed that the classification with the features based on the Triadic measure exhibits the best performance for all classifiers, as also clearly seen in Table 5 with the highest AUC value for this method.

This is not the case for the other networks, as shown in Tables 6 for Kohonen. The GradientBoosting, DecisionTree, and RandomForest classifiers exhibit the best performance using the proposed CN-HA measure, while the LogisticRegression and LinearDiscriminant classifiers exhibit the best performance using node2vec.

The performance of the classifiers for the SmaGri network is similar to that for Political Blogs. Fig. 8 shows ROC for different measures on the LogisticRegression classifier over the WikiVote network, where the proposed SCNHA measure exhibits the best performance.

It can be observed in Table 8 that the proposed SCNHA measure exhibits the best performance on the LogisticRegression and GradientBoosting classifiers, as do CN-AH on the DecisionTree classifier and Triadic on the LinearDiscriminant and RandomForest classifiers. The Triadic measure performs properly in supervised mode, but such motif-based measures exhibit high computational complexity and unpredictable runtimes over large datasets. Our proposed measures, on the other hand, perform well in the supervised mode, while having low time complexity, as stated in Section 3-3.

4.5. *Statistical analysis*

In order to verify whether the obtained differences between different measures are statistically significant, we used the Friedman test (M. Friedman, 1940). For the unsupervised mode, our null hypothesis is that the mean value of Precision and AUC for the fourteen similarity measures are equal. Table 3 to 4 shows the mean and standard deviation of the Precision and AUC along with the P-values obtained from the statistical test. As it can be seen, the P-values are extremely small, indicating that the obtained differences between the similarity measures are statistically significant.

5. Conclusions and future works

In this paper, novel semi-local measures were presented for the link prediction problem on complex networks. We compared the performance of the proposed measures to that of some baseline methods that have been widely used for the link prediction problem. We studied the performance of the methods in both supervised and unsupervised prediction modes. The measures were applied to three real-world networks.

The proposed measures exhibited the best performance in unsupervised mode. For the supervised link prediction problem, we used five well-known classifiers and the *k*-fold cross-validation algorithm. Features based on different measures including those proposed here were used for the classification purpose. One of the proposed measures along with an existing measure showed the best performance in supervised mode. Furthermore, the proposed measures exhibited much lower computational complexity than the others, and can thus be applied to large-scale networks.

Like most similarity measures, our proposed method predicts presence or absence of edges on a directed graph, not the direction of the link. Our future work will be to complement the proposed measure to make them fit to predict not only the existence of the links, but also their direction.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the web. *Social Networks, 25*, 211–230.

Aghabozorgi, F., & Khayyambashi, M. R. (2018). A new similarity measure for link prediction based on local structures in social networks. *Physica A: Statistical Mechanics and its Applications, 501*, 12–23.

Al Hasan, M., Chaoji, V., Salem, S., & Zaki, M. (2006). Link prediction using supervised learning. In SDM06: workshop on link analysis, counter-terrorism and security (Vol. 30, pp. 798-805).

Barabási, A.-L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., & Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications, 311*, 590–614.

Blondel, V. D., Gajardo, A., Heymans, M., Senellart, P., & Van Dooren, P. (2004). A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM review, 46*, 647–666.

Breiman, L. (2001). *Random forests. Machine learning, 45*, 5–32.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and regression trees. Belmont, CA: Wadsworth. *International Group, 432*, 151–166.

Cannistraci, C. V., Alanis-Lobato, G., & Ravasi, T. (2013). From link-prediction in brain connectomes and protein interactomes to the local-community-paradigm in complex networks. *Scientific reports, 3*, 1–14.

Chua, H. N., Sung, W.-K., & Wong, L. (2006). Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions. *Bioinformatics, 22*, 1623–1630.

Dong, Y., Ke, Q., Wang, B., & Wu, B. (2011). In *Link prediction based on local information* (pp. 382–386). IEEE.

Feng, X., Zhao, J. C., & Xu, K. (2012). Link prediction in complex networks: A clustering perspective. *The European Physical Journal B, 85*, 3.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational statistics & data analysis, 38*, 367–378.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics, 11*, 86–92.

Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 855-864).

Grover, N., & Wason, R. (2012). Comparative analysis of pagerank and hits algorithms. *International Journal of Engineering Research & Technology (IJERT), 1*, 1–15.

Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Representation learning on graphs: Methods and applications. arXiv preprint arXiv:1709.05584.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: Concepts and techniques*. Elsevier.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology, 143*, 29–36.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS), 22*, 5–53.

Izenman, A. J. (2013). Linear discriminant analysis. In *Modern multivariate statistical techniques* (pp. 237–280). Springer.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bull Soc Vaudoise Sci Nat, 37*, 547–579.

Jeh, G., & Widom, J. (2002). SimRank: A measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 538–543).

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM), 46*, 604–632.

Laishram, R. (2015). Link Prediction in Dynamic Weighted and Directed Social Network using Supervised Learning.

Leicht, E. A., Holme, P., & Newman, M. E. (2006). Vertex similarity in networks. *Physical Review E, 73*, Article 026120.

Li, J., Peng, J., Liu, S., Ji, X., Li, X., & Hu, X. (2020). Link Prediction in Directed Networks Utilizing the Role of Reciprocal Links. *IEEE Access, 8*, 28668–28680.

Li, S., Huang, J., Zhang, Z., Liu, J., Huang, T., & Chen, H. (2018). Similarity-based future common neighbors model for link prediction in complex networks. *Scientific reports, 8*, 1–11.

Li, Z., Fang, X., & Sheng, O. R. L. (2017). A survey of link recommendation for social networks: Methods, theoretical foundations, and future research directions. *ACM Transactions on Management Information Systems (TMIS), 9*, 1–26.

Liben-Nowell, D., & Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American society for information science and technology, 58*, 1019–1031.

Lichtenwalter, R. N., Lussier, J. T., & Chawla, N. V. (2010). New perspectives and methods in link prediction. In *In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 243–252).

Liu, W., & Lü, L. (2010). Link prediction based on local random walk. *EPL (Europhysics Letters), 89*, 58007.

Liu, Y., Li, T., & Xu, X. (2019). Link Prediction by Multiple Motifs in Directed Networks. *IEEE Access, 8*, 174–183.

Liu, Z., Zhang, Q.-M., Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A local naïve Bayes model. *EPL (Europhysics Letters), 96*, 48007.

Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications, 390*, 1150–1170.

Martínez, V., Berzal, F., & Cubero, J.-C. (2016). A survey of link prediction in complex networks. *ACM Computing Surveys (CSUR), 49*, 1–33.

McGill, M. (1983). Introduction to Modern Information Retrieval McGraw-Hill. New York.

Menard, S. (2002). *Applied logistic regression analysis, (Vol. 106):*. Sage.

Papadimitriou, A., Symeonidis, P., & Manolopoulos, Y. (2012). Fast and accurate link prediction in social networking systems. *Journal of Systems and Software, 85*, 2119–2132.

Pecli, A., Cavalcanti, M. C., & Goldschmidt, R. (2018). Automatic feature selection for supervised learning in link prediction applications: A comparative study. *Knowledge and Information Systems, 56*, 85–121.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. In *In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701–710).

Radmanesh, M., Rezaei, A. A., Al Khafaf, N., & Jalili, M. (2020). In *Topological Deep Network Embedding* (pp. 476–481). IEEE.

Rafiee, S., Salavati, C., & Abdollahpouri, A. (2020). CNDP: Link prediction based on common neighbors degree penalization. *Physica A: Statistical Mechanics and its Applications, 539*, Article 122950.

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N., & Barabási, A.-L. (2002). Hierarchical organization of modularity in metabolic networks. *science, 297*, 1551–1555.

Sarkar, P., Chakrabarti, D., & Moore, A. W. (2011). Theoretical justification of popular link prediction heuristics. *Twenty-Second International Joint Conference on Artificial.* Intelligence.

Schall, D. (2014). Link prediction in directed social networks. *Social Network Analysis and Mining, 4*, 157.

Sørensen, T., Sørensen, T., Sørensen, T., SORENSEN, T., Sorensen, T., Sorensen, T., & Biering-Sørensen, T. (1948). 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.

Tan, F., Xia, Y., & Zhu, B. (2014). Link prediction in complex networks: A mutual information perspective. *PloS one, 9*.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). Line: Large-scale information network embedding. In *In Proceedings of the 24th international conference on world wide web* (pp. 1067–1077).

Tong, H., Faloutsos, C., & Pan, J.-Y. (2006). *Fast random walk with restart and its applications, ICDM'06*, 613–622.

Wang, P., Xu, B., Wu, Y., & Zhou, X. (2015). Link prediction in social networks: The state-of-the-art. *Science China Information Sciences, 58*, 1–38.

Zhang, J., Zhang, Y., Yang, H., & Yang, J. (2014). A link prediction algorithm based on socialized semi-local information. *Journal of Computational Information Systems, 10*, 4459–4466.

Zhou, T., Lü, L., & Zhang, Y.-C. (2009). Predicting missing links via local information. *The European Physical Journal B, 71*, 623–630.