

11-1-2022

Hill Climbing-Based Efficient Model for Link Prediction in Undirected Graphs

Haji Gul

Institute of Management Sciences

Feras Al-Obeidat

Zayed University, feras.al-obeidat@zu.ac.ae

Adnan Amin

Institute of Management Sciences

Fernando Moreira

Universidade Portucalense Infante D.Henrique

Kaizhu Huang

Duke Kunshan University

Follow this and additional works at: <https://zuscholars.zu.ac.ae/works>



Part of the [Computer Sciences Commons](#)

Recommended Citation



Gul, Haji; Al-Obeidat, Feras; Amin, Adnan; Moreira, Fernando; and Huang, Kaizhu, "Hill Climbing-Based Efficient Model for Link Prediction in Undirected Graphs" (2022). *All Works*. 5506.

<https://zuscholars.zu.ac.ae/works/5506>

This Article is brought to you for free and open access by ZU Scholars. It has been accepted for inclusion in All Works by an authorized administrator of ZU Scholars. For more information, please contact scholars@zu.ac.ae.

Article

Hill Climbing-Based Efficient Model for Link Prediction in Undirected Graphs

Haji Gul ^{1,†}, Feras Al-Obeidat ^{2,*,†}, Adnan Amin ^{1,†} , Fernando Moreira ^{3,4,†}  and Kaizhu Huang ^{5,†}¹ Center for Excellence in Information Technology, Institute of Management Sciences, Peshawar 25000, Pakistan² College of Technological Innovation, Zayed University, Abu Dhabi 144534, United Arab Emirates³ REMIT, IJP, Universidade Portucalense, 4200-072 Porto, Portugal⁴ IEETA, Universidade de Aveiro, 3810-193 Aveiro, Portugal⁵ Data Science Research Center, Division of Natural and Applied Sciences, Duke Kunshan University Duke Avenue No. 8, Kunshan, Suzhou 215316, China

* Correspondence: feras.al-obeidat@zu.ac.ae

† These authors contributed equally to this work.

Abstract: Link prediction is a key problem in the field of undirected graph, and it can be used in a variety of contexts, including information retrieval and market analysis. By “undirected graphs”, we mean undirected complex networks in this study. The ability to predict new links in complex networks has a significant impact on society. Many complex systems can be modelled using networks. For example, links represent relationships (such as friendships, etc.) in social networks, whereas nodes represent users. Embedding methods, which produce the feature vector of each node in a graph and identify unknown links, are one of the newest approaches to link prediction. The Deep Walk algorithm is a common graph embedding approach that uses pure random walking to capture network structure. In this paper, we propose an efficient model for link prediction based on a hill climbing algorithm. It is used as a cost function. The lower the cost is, the higher the accuracy for link prediction between the source and destination node will be. Unlike other algorithms that predict links based on a single feature, it takes advantage of multiple features. The proposed method has been tested over nine publicly available datasets, and its performance has been evaluated by comparing it to other frequently used indexes. Our model outperforms all of these measures, as indicated by its higher prediction accuracy.

Keywords: complex network analysis; local link prediction methods; link prediction; complex networks; hill climbing

MSC: 05C82

Citation: Gul, H.; Al-Obeidat, F.; Amin, A.; Moreira, F.; Huang, K. Hill Climbing-Based Efficient Model for Link Prediction in Undirected Graphs. *Mathematics* **2022**, *10*, 4265. <https://doi.org/10.3390/math10224265>

Academic Editor: Alfonso Niño

Received: 30 June 2022

Accepted: 27 August 2022

Published: 15 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The development of network information technology has led to the emergence of many complex systems. As a result, it is essential to identify and characterise these complex systems from both a theoretical and practical aspect [1,2]. Complex network theory is now a useful tool for studying a variety of complex systems [3–6], including social networks, protein–protein interaction networks, and so on. A social network is made up of a group of social actors and their interactions. It can be seen as a graph, with nodes representing the actors and edges representing the connections between them. Relationships are generally formed between people based on their shared interests. Behaviour is created when new nodes and relationships are added to a social network. It is important to predict the likelihood of a relationship forming between pairs of nodes in the context of social network research and mining. The “link prediction problem” is a well-known example of this. If a social network has an immediate image at time t , link prediction aims to estimate the edges that are likely to form in the time between t and $t + 1$ [7]. The suggestion of friends, movie

and music recommendations based on users' backgrounds are examples of link prediction applications in online social networks.

There are several uses for the problem of link prediction. For example, link prediction can be used to differentiate research areas in scientific publications [8] and in spam mail detection [9], recommendation systems [10], social network privacy control [11], disease prediction [12], network reconstruction [13,14], expert detection [15], partially labelled network classification [16], network evolving mechanisms evaluation [14] and protein–protein interaction prediction [17]. The challenges associated with business-to-business associations [18] can also be resolved using LP.

For the solution of link prediction, various approaches have been presented. Some of these methods work on structural similarities between nodes. Measures based on local and quasi-local similarity have recently gained a lot of interest as a way to solve this problem. This is due to their simplicity while still being useful and relatively high prediction accuracy. The majority are determined by node degree and common neighbors [19]. Furthermore, traditional local measures such as common neighbors, Adamic Adar and resource allocation ignore the neighborhood's directions. Such metrics do not differentiate between directed and undirected graphs.

Aghabozorgi and Khayyambashi [20] proposed a local measure of similarity between pairs nodes that includes not only the number of common neighbors but also other structural information. This metric only considers triadic blocks and neglects larger ones. Additionally, this approach depends on network motifs, which are computationally expensive and inappropriate for large graphs. Pecli et al. published on a supervised technique [21]. They demonstrated how their feature selection methodology increases classifier performance. A similarity metric based on future common neighbors that categorizes neighbors into three groups was proposed by S. Li et al. [22]. Haji et al. [23] introduced a novel technique based on a double degree equation with a network feature for similarity calculation. Gao et al. [24] introduced a similarity method based on linear dynamical response. Clauset et al. [25] designed a hierarchical structure-based technique that employs the network hierarchy to calculate the likelihood of a link. Zhu et al. [26] have produced an information-theoretic model making use of complex network topology to estimate links. Pech et al. [27] planned a robust principal component analysis-based (robust PCA) method to identify the missing link in a complex network.

There are many link prediction techniques based on global, semi-local or local network properties. However, they usually show inconsistent performance over large networks. Additionally, the dataset imbalance caused by a small number of known links and a large number of unknown links affects machine learning classifier-based link prediction algorithms. Neural network-based approaches also need strong resources and a lot of time to execute [28]. In order to determine the similarity between pairs of nodes, x and y , and improve the accuracy, we designed a novel approach for link prediction in a complex network. Assume that x and y are two nodes in a complex network that is undirected and unweighted and that $G = (x, y)$. The likelihood of two nodes x and y connecting in the future is defined by their similarity rate at time t based on previous information about the nodes. The proposed method makes use of the hill climbing approach as well as the complex network's quasi and local properties. The goal is to improve the accuracy by including local topological features. The hill climbing approach computes the cost function of each topological feature. Prediction accuracy increases with decreasing cost function. The proposed method's results have been compared to those of baseline algorithms.

Our primary contributions include the following.

- We introduce a different approach to link prediction based on hill climbing criteria with quasi and local complex network features. The proposed method computes the cost function value, and the lower cost function provides a higher accuracy result.
- We conduct studies on many complex networks of various sizes and structures, assessing the different link prediction approaches.

Motivation

The following aspects inspired the research:

One of the most popular study topics in the area of complex network analysis is link prediction. Several useful applications could arise from this issue, including recommender systems [29,30], estimated high influence users in a collaboration network [31] and disease prediction [12]. Better decision models with higher performance values and fewer market risks can be constructed based on the observed patterns of structural business changes.

It has been recognized that numerous authors have used similarity indexes to predict future linkages [32–37]. In their algorithms, they put out one of the topological-based similarity indices. Many similarity metrics have been put out based on various features of the network's topological information. As a result, an algorithm may perform better in one similarity index while not performing significantly well in other similarity measures for the same network. As a result, algorithm performance differs from network to network. This encourages us to overcome the link prediction problem by employing a number several similarity indexes.

Advantages:

- The proposed work simultaneously makes use of many topological properties.
- In order to overcome the shortcomings of other algorithms, the revised hill climbing base solution is offered. In each state, it selects the lowest-cost topological feature for link prediction.
- Compared to other prediction indexes, it improved link prediction accuracy.

Disadvantages:

- The proposed approach has only been evaluated on undirected networks, ignoring directed and weighted networks
- The proposed work also requires longer execution times for large networks, similar to other algorithms.

The rest of the article is structured as follows: Section 2 presents details about the related work of link prediction; Section 3 outlines experimental setup, proposed methodology details and comparative analysis. Section 4 presents results and discussions. While the final section, includes a conclusion and references.

2. Preliminary

A brief overview of previous link prediction research is provided. Many survey papers for link prediction have previously been published [38,38–40]. In this paper, these approaches are mainly divided into three categories: similarity, embedding and probabilistic-based link prediction.

2.1. Similarity-Based Methods

The similarity is often expressed using the terminology “proximpair”. For each pair of nodes x and y , a score s_{xy} is considered in these algorithms. Each prospective link is given a score, and the greater the score, the more probable the link will be stable between both the nodes. Similarity measures are more applicable in realistic scenarios for large-scale networks due to their high computational load. A broad framework for the link prediction problem was proposed by Wang et al. [40]. This framework encompasses two sorts of methodologies: unsupervised and supervised machine learning similarity methods. Many existing approaches rely on some form of node-to-node similarity estimate. The next sections consist of many link prediction discussions with formulations that have been proposed in the literature.

Common Neighbor (CN) [7]: Given in Equation (1) is a local similarity-based method. For the measurement of similarities, this metric employs the intersection of the common

neighbors of the pairs of nodes. The s_{xy} similarity between two nodes x and y is calculated as follows:

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

where $\Gamma(x)$ is the set of node x 's neighbors. Despite its simplicity, this measure has a high degree of accuracy on most real-world complex networks [19].

Jaccard's Coefficient (JC) [41]: It is the normalized form of a common neighbor, which computes similarity based on a common neighbor with a set of neighbors from x source to y destination node as given in Equation (20).

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

Adamic Adar (AA) [42]: Initially, it was introduced for calculating the similarity between two web pages. The mathematical form is expressed in Equation (21). In this similarity measure, the higher score is assigned to the pairs of nodes having the least number of common neighbors.

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

Resource allocation (RA) [43]: This metric was derived from the physical resource allocation method. It has a strong similarity to AA. Both of these methods penalize common neighbors to a large extent, but RA is more precise than AA in this regard. The following formula given in Equation (22) is used to determine the metric:

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

AA and RA frequently have similar prediction accuracy for networks with lower average degrees, but RA is more efficient for networks with large average degrees [44].

Preferential Attachment (PA) [45]: In many real-world networks, node degree distribution follows a power-law pattern. The existence of scale-free networks is owing to this property. In Equation (23), following is the definition of the metric:

$$PA(x, y) = |\Gamma(x) \cdot \Gamma(y)| \tag{5}$$

Salton Index: This is referred to as the cosine measure and is related to the Jaccard coefficient. It is defined as follows, Equation (6):

$$SI(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma_x| * |\Gamma_y|}} \tag{6}$$

Hub Promoted Index (HPI) [46]: Ravasz proposed this measure in 2002 to investigate the metabolic network's modular structure. This type of network has a hierarchical structure with small internal modules that are all independent of one another. The metric is described as follows given in Equation (7). Its major goal is to prevent linkages from being established primarily between hubs and to promote the construction of links between nodes with low degrees and hubs.

$$HPI(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{\min(|\Gamma(x)|, |\Gamma(y)|)} \tag{7}$$

Leicht Holme Nerman: The LHN index assigns a high degree of similarity between node pairs that share common neighbors [47]. This approach was put forth to evaluate

the similarity between nodes in a real-world complex network. The index is represented mathematically in Equation (8).

$$LHN(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cdot \Gamma(y)|} \quad (8)$$

Parameter Dependent: It was created by Zhu et al. [48] and accurately calculates the similarity. It degenerates to CN, LHN and SI index if the free parameter is equal to ($\lambda = 0$, $\lambda = 1$ and $\lambda = 0.5$). If the free parameter is zero ($=0$), it degenerates to CN. The mathematical formula can be expressed as follows:

$$PD(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{(|\Gamma(x) \cdot \Gamma(y)|)^\lambda} \quad (9)$$

2.2. Embedding-Based Methods

Node embedding is also used to calculate the similarity between pairs of nodes [49]. As a result, to address link prediction, some popular embedding methods are matrix factorization [50], stochastic block [51] and so on. Recent improvements such as deep-walk [52], LINE [53] and node2vec [54] have been proposed to train node embedding via the skip-gram method, inspired by word embedding methods in natural language processing applications. Deep-walk generates random walks of a given length for each vertex and selects the next visited node from the current node's neighbors. The skip-gram approach is used to discover node embedding from the resulting node sequence. To reconstruct the complex network structure, a graph auto-encoder (GAE) [55] is proposed to train the nodes embedding using graph convolution neural networks. The node embedding methods can extract useful information from the graph and so perform well in the link prediction challenge. However, if the graph gets exceedingly sparse, the efficiency of link node embedding approaches can suffer.

2.3. Probabilistic-Based Method

The probability of link establishment is calculated after a series of rules are extracted using maximum likelihood estimation methods. Probabilistic approaches produce an abstract model of the network through which link predictions can be made. Maximum likelihood estimations and probabilistic models generally outperform similarity-based techniques in terms of accuracy, although they increase the time complexity [14].

A survey on social networks was undertaken by P. Kalpana et al. [28]. They claim that compared to previous methodologies, the deep convolution neural network produces better results. Such a thing is implemented based on the knowledge obtained from the information obtained from the many social networks. Deep-learning-based algorithms are employed to make better predictions about the future. Predicting future links and node similarity in large-scale networks still presents significant difficulties. In general, neural network-based link prediction techniques produce accurate results but require a lot of resources and take longer executing time.

Due to a lack of labels for training, the supervised method's prediction accuracy can sometimes suffer [56]. Unsupervised link predictors, however, continue to be inflexible and domain specific. The majority of topological algorithms predict association on the basis of node information and qualities. The accuracy of the predictions can be impacted by the fact that not all of the nodes' information is widely accessible to the public.

According to Victor M et al. [19], link prediction continues to be an open research challenge, given its significance in various applications. In most networks, it has been found that new links can be predicted more accurately using solely local or quasi-local information. Additionally, only a few strategies can adapt to the network's global structure, and none can adapt to its local structure. The size of complex networks poses the biggest challenge in practise because it affects the kind of approaches that may be used.

Therefore, a solution based on local and quasi-local network information needs to be developed in this context for link prediction. Here, it is best to use a hill climbing strategy to take advantage of various local and quasi topological characteristics. The method proposed in this paper is an updated hill climbing approach. The novelty of this work is that it compares the cost functions result of different topological features on the current state. However, the original hill climbing algorithm compares the current state result with the previous state. The details are given in the methodology section.

3. Materials and Methods

The following is a description of a frequent evaluation metric that is used to compare the proposed technique to other LP approaches.

3.1. Evaluation Metric

Link prediction technology can be evaluated using a variety of techniques. In this work, AUC [14,57] is used to assess link prediction techniques that take results from various angles view. The best and most used metric for assessing link prediction techniques is AUC. The AUC value is: If n is the total number of executions, n' is the number of times a missing link has a better result than a non-observed link, and n'' is the number of times where a missing link has a similar result to a non-observed link. Then the AUC is defined as given in Equation (10).

$$AUC = \frac{n' + 0.5n''}{n} \quad (10)$$

Consider the undirected and unweighted network $G = (V, E)$. The identified links are referred to as the set E . In the complex network, let E' deal with the non-existent links. If U refers to all possible $\frac{|V|(|V|-1)}{2}$ links that make up a network G , then $E' = U \setminus E$. The observed links, E , are randomly isolated into two independent sets, specifically, a test set E^P and a training set E^T , to analyse the performance of link prediction algorithms. The data in E^T is used to predict missing links, whereas the data in E^P is used to evaluate how well the prediction performed. The metric AUC is defined in this article as the probability that a randomly chosen edge in E^P would capture a higher score than a randomly chosen edge in E' .

3.2. Datasets

To test the performance of proposed and alternative methods, we employed nine publicly available benchmark datasets from various fields. Table 1 emphasises the statistics of the dataset used in this study, where N denotes the total number of nodes, E total number of links, $MaxD$ maximum degree, $Avg G$ average degree and $Delta$ density. This page's footnote has a download link: <https://networkrepository.com/>, accessed on 28 June 2022, <https://snap.stanford.edu/data/>, accessed on 28 June 2022, <http://konect.cc/networks/>, accessed on 28 June 2022. These are all unweighted and undirected datasets. The following are brief descriptions of the datasets that were used.

Elegans (neuron) [58]: There are 279 neurons and 2990 links in this dataset, with 1584 unidirectional and 1406 bidirectional links. The direction was ignored in our studies, resulting in a total of 2287 linkages.

US Air [59]: A network of 500 airports in the United States with direct flights. The nodes represent airports, and two nodes are connected if the associated airports have a direct flight.

Router [60,61]: A router network is a complex network that lacks weight and direction. It is essentially an Internet network in which the routers are represented by nodes and the connections between them by links.

US Power Grid [60]: This network contains data about the western United States of America's power grid. A generator, transformer or substation is used to represent each node, meanwhile a power line is used to represent each edge.

Yeast [62]: This is a commercial leaving agents network made up of nodes (yeast cells) and links (interactions).

Karate [63]: It is a unidirectional and unweighted human karate club complex network dataset in which nodes/vertices represent individuals and edges/links represent interactions between individuals or karate club members.

Dolphin [60]: Dolphins express dolphin network nodes/vertices, and links/edges indicate the relationships between two dolphins in an undirected animal complex network. Weight and self-loops are not taken into account.

Hamster Full [64]: It is a social complex network that is both unweighted and undirected. Users of the website hamsterster.com are linked through friendships and family ties in this network.

Route Views [64,65]: It is the Internet’s undirected, interconnected network of autonomous systems. Edges represent communication, while nodes represent autonomous systems (AS).

Table 1. Datasets Statistics.

Datasets	N	E	MaxD	Avg D	Δ
Elegans [58]	279	2287	268	16.394	0.059
US Air [59]	500	2980	139	11.92	0.024
Routers [61]	3722	6258	103	2.493	0.00091
US Power Grid [60]	4939	6594	19	2.67	0.00054
Yeast [62]	1458	1948	56	2.7	0.0.0018
Karate [63]	33	78	17	4.5882	0.1477
Dolphin [60]	62	159	12	5.1290	0.0841
Hamsters [64]	2426	16631	273	13.710	0.0057
RouteViews [65]	6474	13,895	1459	4.293	0.00067

3.3. Methodology

Quasi and local properties such as local random walk, local path, similar neighbors, adamic adar, resource allocation, jaccard, preferential and hub promoted among others, are included in a complex network. They are all based on quasi and local topological features and work as quasi or local similarity criteria. For each pair of nodes x and y , the performance of each global, quasi and local feature similarity technique varies. This is due to the structure of complex networks’ lack of method-required features. If the number of common neighbors between node x and the targeted node y is greater, the CN method assigns a higher score. Similarly, in adamic adar and resource allocation, higher accuracy is assigned if common neighbors are fewer in number between x and y . How can the similarity based on a neighbor be calculated if there is not one? Now here is the need for a model to be created to overcome this failure and improve the result.

In this paper, we introduce a method based on an extended form of the hill climbing algorithm. Hill climbing is a mathematical optimization method that is part of the local search category. It is an iterative method that starts with a random solution to a problem and then tries to improve it by making small changes to the result. If the change results in a better solution, another progressive adjustment is made to the new solution, and so on until no more enhancements are possible. This is the actual working procedure of the hill climbing algorithm. Hill climbing (HC) and random walking (RW) are processes that are somehow comparable. When it comes to selecting the next node, there is a slight difference between HC and RW. RW chooses a node randomly, whereas HC computes the resultant value using the cost function given in Equation (11).

$$HC\ CostFunction = (1 + e^{(-\Delta E)})^{-1} \tag{11}$$

$$\begin{aligned} &\because (-\Delta E) = x - y \text{ or } n - c \\ &(x = c = current, y = n = next) \end{aligned} \tag{12}$$

$$\therefore HC\ CostFunction = (1 + e^{-(x-y)})^{-1} \tag{13}$$

The amount of change between the current and next node is denoted by ΔE in Equation (11). It can be written as $x - y$ (current and next node), and based on it Equation (13) has been updated. The numerical constant e has the value 2.7182.

The proposed method work as follows: first, it selects the node (selection of source node x), and then it computes the cost of each method using a hill climbing cost function. The greater the similarity between the source and targeted nodes (y), the lower the cost values (calculated with a cost function) will be. The feature with the highest similarity and lower cost value has been considered and other features omitted only for the current node. In the second pass to predict links in other pairs of nodes, it now chooses the next node and the best quasi-local and local similarity feature technique for the next prediction. Similarity: this approach was performed for the entire network. The following steps comprise the proposed method:

- **Selection of Source Node:** The source node x has been selected following the dataset sequence.
- **Backtrack-less:** At this stage, various quasi and local feature algorithms are applied to identify the targeted node y for source node x . Usually, each algorithm works based on one or two complex network features. These features or techniques are used to find similarity, and then the cost function is used to compute the cost value for each feature. All the cost function results are then embedded in a vector and compared with each other. Finally, identified the lowest cost and highest similarity feature. As a result of a higher similarity (between pairs of nodes x and y) and a low-cost feature/technique, the targeted node has been identified (link predicted). This phenomenon is called “backtrack-less walk”. Because it does not move toward the last node to compare it to the present node, which is the pure workflow of hill climbing, this is the major difference between hill climbing and our proposed work, presented in this paper for link prediction.
- **Incremental:** Once the targeted node y has been discovered using the lowest cost feature(s) algorithm, we move to the next node. Each node is expressed with a numerical value in the datasets, such as $1, 2, 3, \dots, n$. The next node is thus chosen randomly. This approach has been followed to predict links across the whole network.

To begin, we compute the similarity of local and quasi-local techniques between the source x and the target node y . Then we evaluate each method using AUC and find the final value of each method. Next, we used a hill climbing cost function to compute the cost values of each method’s final result. For example, if there are n number of mutual nodes, which is a single feature similarity estimation method, we need to compute this for the link prediction problem. The mathematical formula can be written as given in Equation (14):

$$HC\ CostFunction = (1 + e^{(|\Gamma(x) \cap \Gamma(y)|)})^{-1} \tag{14}$$

More simply, the given Equation (14) can be re-written as Equation (16):

$$\therefore \delta_1(x, y) = |\Gamma(x) \cap \Gamma(y)| \tag{15}$$

$$\therefore HC\ CostFunction_1 = (1 + e^{(\delta_1)})^{-1} \tag{16}$$

For more than one feature technique that works on mutual nodes, such as salton, the source and destination nodes agree under the square root, given in Equations (17) and (19). The cost function formula can be written as:

$$HC\ CostFunction = (1 + e^{(\frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma_x| * |\Gamma_y|}})})^{-1} \tag{17}$$

$$\therefore \delta_2(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma_x| * |\Gamma_y|}} \tag{18}$$

$$\therefore HC\ CostFunction_2 = (1 + e^{(\delta_2)})^{-1} \tag{19}$$

We have included six structured-based techniques in our work. Similarly, the cost functions for this jaccard, Adamic-adar, resource allocation, preferential attachment, LHN and PD can be computed using Equations (20)–(23).

$$HC\ CostFunction_3 = (1 + e^{(\frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|})})^{-1} \tag{20}$$

$$HC\ CostFunction_4 = (1 + e^{(\sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log|\Gamma(z)|})})^{-1} \tag{21}$$

$$HC\ CostFunction_5 = (1 + e^{(\sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\Gamma(z)})})^{-1} \tag{22}$$

$$HC\ CostFunction_6 = (1 + e^{(|\Gamma(x) \cdot \Gamma(y)|})})^{-1} \tag{23}$$

$$HC\ CostFunction_7 = (1 + e^{(\frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cdot \Gamma(y)|})})^{-1} \tag{24}$$

$$HC\ CostFunction_8 = (1 + e^{(\frac{|\Gamma(x) \cap \Gamma(y)|}{(|\Gamma(x) \cdot \Gamma(y)|)^\lambda})})^{-1} \tag{25}$$

The generic formula of the HC cost function can be expressed as follows:

$$HC\ CostFunction_{gen} = \sum_1^n (1 + e^{evaluated-result})^{-1} \tag{26}$$

Figure 1 depicts the proposed framework in graphic form, while the pseudo-code is given in Algorithm 1. The dataset was first imported, and an adjacency matrix, which is a representation of a complex network graph, was produced. As we already know, a graph can be represented through an adjacency matrix or an adjacency list. The adjacency matrix is then separated into two categories: training E^T and testing E^P . The percentage of K-folding is 10%. All algorithms have been tested on E^P after they have been trained on E^T and moved to the trained module. We then compute the cost value after analyzing all of them after evaluating the predicted matrix received for each approach. For comparisons, all cost function results are saved in a vector. Finally, the best lower-cost technique with the highest accuracy in the link prediction problem has been discovered. Based on the specified criteria, the next node has been discovered. The code is available through the link <https://github.com/hajigul/Efficient-LP-Method-Code>, accessed on 28 June 2022.

Algorithm 1 Hill-Climbing-Based Link Prediction

Require: Modified Set $E^U - E = E^{ProbSet}$

Ensure: Lowest Cost Value

- 1: **for** $k = 1..n$ **do**
 - 2: Compute Topological Features Result
 - 3: **for** $s = 1..m$ **do**
 - 4: Compute Cost Value of the Feature
 - 5: Generate a List of Cost Functions
 - 6: $s_v = \min(\min(\text{List of Cost Functions}))$
 - 7: Reported the Lowest One
 - 8: **end for**
 - 9: $k = k + 1$
 - 10: **end for**
-

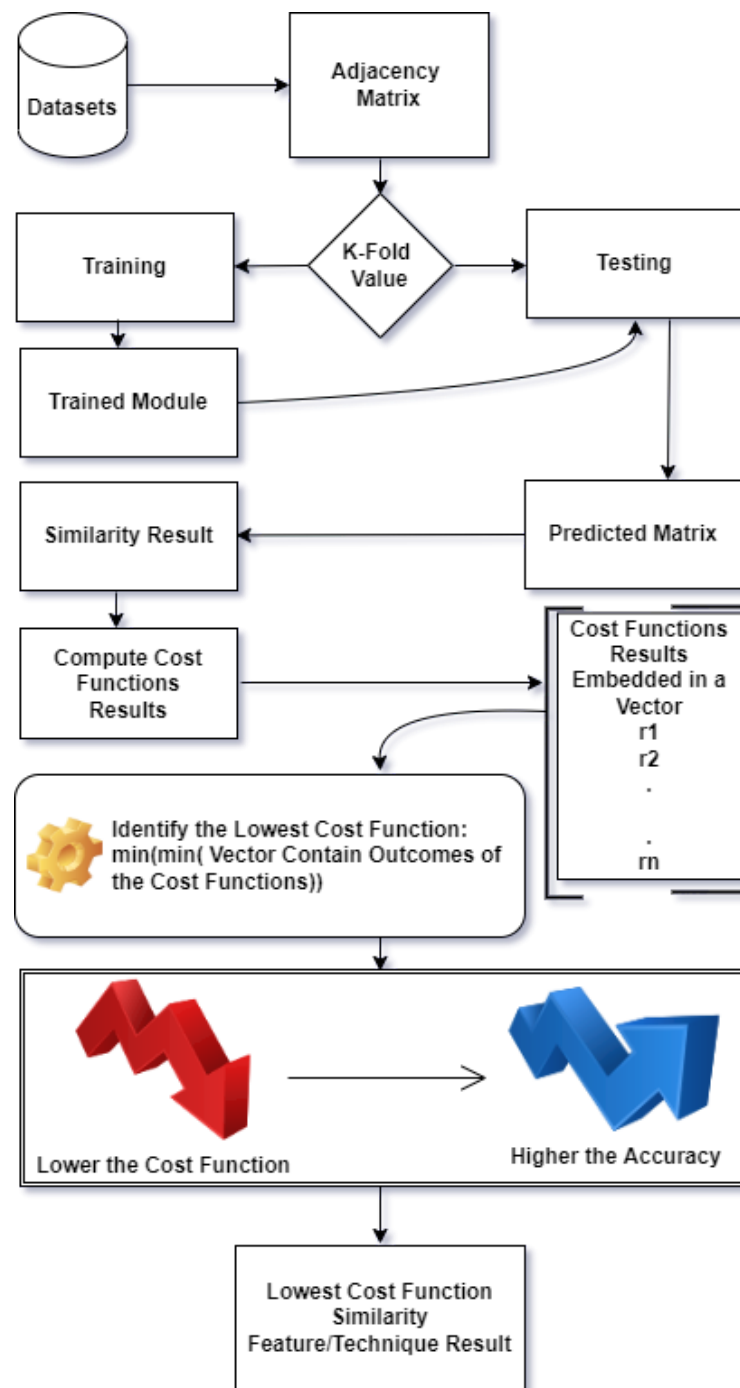


Figure 1. Methodology.

3.4. Comparative Analysis

Table 2 compares the proposed technique’s performance to that of other well-known approaches for link prediction problems. Over the different complex network datasets listed in Table 2, the proposed technique enhanced accuracy by 2.45%, 4.08%, 0.75%, 2.9%, 1.1% and 1.2%. The way that various algorithms operate and produce distinct AUC values due to the irregular structure and properties of complex networks global topological methods only function on global features, whereas local topological methods only function on local features. Strong local or global features are present in some complex networks, and these features have an obvious impact on the AUC result. In general, the complex network structure is directly correlated with the AUC findings.

The letter “X” appears in various columns in Table 2. It indicates that this dataset was not used in the experiments in the publication. Assume that “X” is present in the second column against double-degree paper, node2vec, line, sdne, care, celp and calp. The elegans dataset was not used by the authors of these studies. For the same reason, “X” is provided in additional columns.

Table 2. Comparative analysis of proposed technique with other well-known approaches.

Previous Work	Elegans	Routers	US Power	Yeast	Karate	Dolphin
Double-Degree [52]	X	65.07%	85.97%	83.99%	X	X
Node2Vec [66]	X	58.98%	85.98%	78.95%	X	X
LINE [53]	X	67.03%	82.09%	85.97%	X	X
SDNE [67]	X	65.52%	84.03%	84.05%	X	X
CARE [66]	X	65.28%	89.65%	88.59%	X	X
CELP [66]	X	68.88%	91.08%	90.68%	X	X
CALP [66]	X	70.99%	92.27%	91.77%	X	X
LO [68]	67.51%	X	X	80.16%	63.82%	73.04%
CND [69]	85.79%	X	X	80.16%	63.82%	73.04%
Proposed	88.24%	82.03%	93.02%	94.97%	78.01%	82.65%

4. Results and Discussions

We randomly separated the set of observed links in the network, E , into two sets, a training set E^T and a probe set E^P in order to analyze the accuracy of the link prediction methods and compare it to the performance of the proposed technique. In the first attempt, the training set had 90% of the observed linkages, while the probe set contained 10%. The accuracy of all the methods was tested using the identical training and probe sets ratio. The experiment process was repeated 100 times, with a random sampling of the observed connections conducted in each run. Table 3 summarises the average prediction accuracy of all 100 runs. The results are represented graphically in Figure 2.

Table 3. Common neighbor (CN), Jaccard coefficient (JC), Adamic-adar (AA), resource allocation (RA), preferential attachment (PA), and (SI) stand for Sorensen index.

Datasets	CN	JC	AA	RA	PA	SI	HPI	LHN	PD	Proposed
Elegans	0.8312	0.8007	0.8671	0.8701	0.7512	0.8093	0.8389	0.7101	0.6883	0.8824
US Air	0.9423	0.9202	0.9631	0.9572	0.7682	0.9201	0.8998	0.8002	0.7624	0.9581
Routers	0.5708	0.6169	0.7161	0.7795	0.7204	0.6859	0.7154	0.5832	0.6656	0.8203
US Power Grid	0.5927	0.6354	0.9083	0.8913	0.7108	0.6308	0.6885	0.7491	0.6955	0.9302
Yeast	0.9086	0.7105	0.9206	0.9108	0.7035	0.9076	0.9207	0.7864	0.6553	0.9497
Karate	0.7167	0.6253	0.7502	0.7691	0.7233	0.6624	0.7005	0.5965	0.6237	0.7801
Dolphin	0.8103	0.7587	0.8021	0.0.8145	0.7355	0.6878	0.6991	0.7658	0.5896	0.8265
Hamsters	0.7502	0.7658	0.7789	0.7854	0.6254	0.6999	0.7125	0.5471	0.6785	0.8001
Route Views	0.8088	0.7165	0.7953	0.0.6987	0.6214	0.5844	0.6963	0.5896	0.6987	0.8399

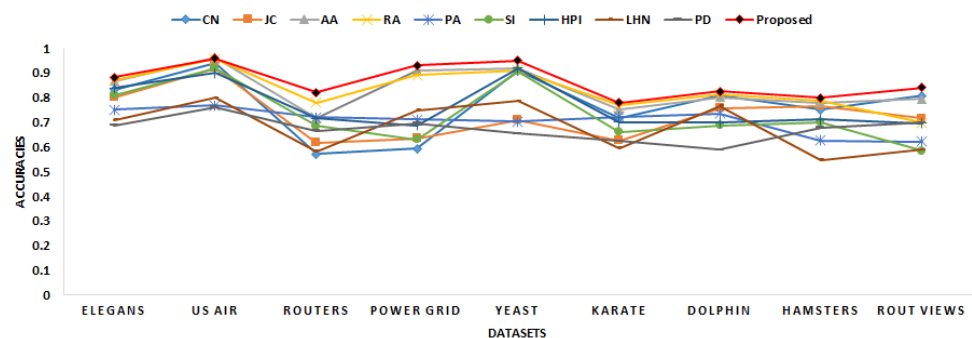


Figure 2. Accuracy over different datasets.

We used a total of nine datasets in the experiments, each with a different size and structure. Over these 9 datasets, 10 link prediction algorithms, including the proposed one, were assessed. Compared to existing techniques, the proposed approach provides a greater accuracy over six. While compared to all other approaches, Adamic-adar has a greater accuracy over one dataset (US air). Over various datasets, each method performs differently. It is due to the structure and characteristics of complex networks.

We test the prediction performance with different partitioning sizes of training and probe sets to further study the performance of the proposed technique and compare it to state-of-the-art methods. We picked different sizes of probe sets for this purpose: 20%, 30%, 40% and 50%, correspondingly. For this experiment, we used nine complex network datasets. In Figure 3, we show the average accuracies of 100 separate runs of each experiment (with a random samples division of E into E^P and E^T) to visualize and measure the findings. There in the plot of Figure 3, we have also included the findings from the previous experiment in which we used a probe set size of 10% for comparative purposes. It is worth noting that the time it takes to compute AUC for big networks rises exponentially as probe size grows. As a result, we did not include the hamster and route views datasets in this experiment.

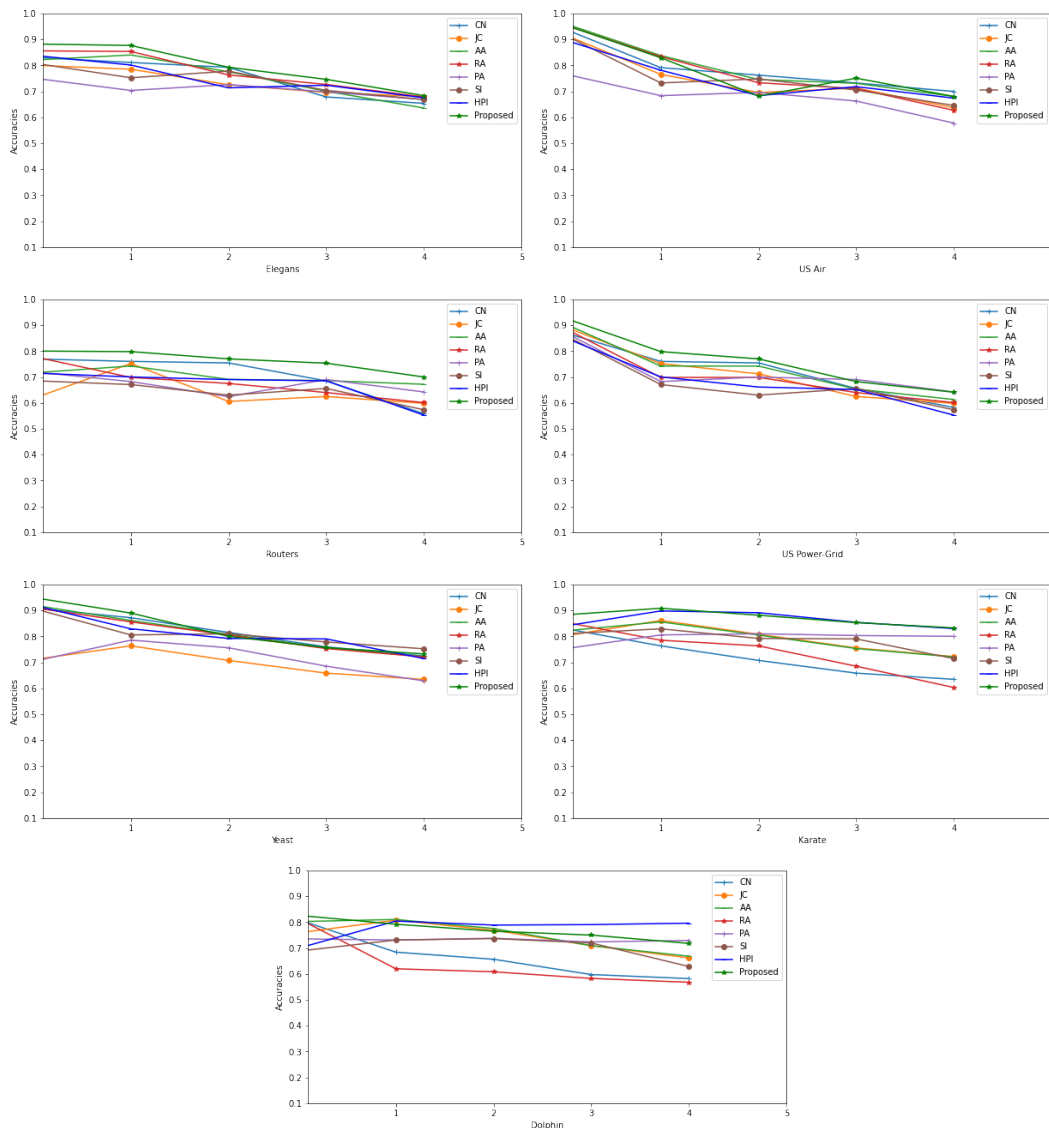


Figure 3. 10%, 20%, 30%, 40% and 50% ration results of each dataset.

5. Conclusions

In the field of complex network analysis, link prediction is a developing research area. It could be used to study and analyze the development of social groupings within a network. Such analysis can help us find hidden groups or the relationships that are missing in the groupings in the appropriate use models. Forming a recommendation system is one of the common applications of link prediction, which is clearly useful in modelling a successful business plan. This is commonly used in various real-world applications, specifically in e-commerce. Data analysis for security and crime prevention investigations could be another use for link prediction. With the help of these link prediction models, the possible threat within the terrorist network can also be recognized. We propose a method in this paper that is based on a hill climbing approach and complex networks of quasi-local and local properties. Other local or quasi-methods to predict links based on a single feature may not be able to predict links if the feature is unavailable, e.g., no common neighbor or appropriate node degree. In contrast, hill climbing computes the cost function result of each topological feature. Prediction accuracy increases with decreasing cost function results. The lower the cost value is, the higher the prediction result will be. Compared to other state-of-the-art procedures, the accuracy has greatly improved with the proposed method. In this paper, only undirected and unweighted networks have been used. However, the proposed similarity indices can easily be extended to more complicated cases, such as directed or weighted networks. It would also be great from a theoretical viewpoint to combine more local and path-based indices that can be used to predict links more accurately.

Author Contributions: Conceptualization, H.G., A.A., F.A.-O., K.H. and F.M.; methodology, H.G., A.A. and F.A.-O.; investigation, H.G., A.A., F.A.-O., K.H. and F.M.; resources, F.A.-O., K.H. and F.M.; writing—original draft preparation, H.G. and A.A.; writing—review and editing, H.G., F.A.-O. and A.A.; supervision, A.A.; funding acquisition, F.A.-O., K.H. and F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Ethics Committee of Ethical review and approval were waived for this study, due to REASON Not applicable for studies not involving humans or animals.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Urena, R.; Kou, G.; Dong, Y.; Chiclana, F.; Herrera-Viedma, E. A review on trust propagation and opinion dynamics in social networks and group decision making frameworks. *Inf. Sci.* **2019**, *478*, 461–475. [[CrossRef](#)]
2. Coquidé, C.; Georgeot, B.; Giraud, O. Distinguishing humans from computers in the game of go: A complex network approach. *EPL (Europhys. Lett.)* **2017**, *119*, 48001. [[CrossRef](#)]
3. Wu, M.; Ye, Z.; Wen, X.; Jiang, X. Air traffic complexity recognition method based on complex networks. *J. Beijing Univ. Aeronaut. Astronaut.* **2020**, *46*, 839–850.
4. Bodaghi, A.; Goliaei, S.; Salehi, M. The number of followings as an influential factor in rumor spreading. *Appl. Math. Comput.* **2019**, *357*, 167–184. [[CrossRef](#)]
5. Zengping, Z.; Yu, H.; Kuanyun, G.; Bo, L.; Yinghao, Z. Scale-free Characteristics and Link Prediction in Complex Railway Network. *J. Phys. Conf. Ser.* **2021**, *1955*, 012099. [[CrossRef](#)]
6. Sahhaf, S.; Tavernier, W.; Papadimitriou, D.; Careglio, D.; Kumar, A.; Glacet, C.; Coudert, D.; Nisse, N.; Fàbrega, L.; Vilà, P.; et al. Routing at large scale: Advances and challenges for complex networks. *IEEE Netw.* **2017**, *31*, 108–118. [[CrossRef](#)]
7. Liben-Nowell, D.; Kleinberg, J. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.* **2007**, *58*, 1019–1031. [[CrossRef](#)]

8. Gallagher, B.; Tong, H.; Eliassi-Rad, T.; Faloutsos, C. Using ghost edges for classification in sparsely labeled networks. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 24–27 August 2008; pp. 256–264.
9. Huang, Z.; Zeng, D.D. A link prediction approach to anomalous email detection. In Proceedings of the 2006 IEEE International Conference on Systems, Man and Cybernetics, Taipei, Taiwan, 8–11 October 2006; IEEE: Piscataway, NJ, USA, 2006; Volume 2, pp. 1131–1136.
10. Kaya, B. A hotel recommendation system based on customer location: A link prediction approach. *Multimed. Tools Appl.* **2020**, *79*, 1745–1758. [[CrossRef](#)]
11. Dhannuri, S.P.; Sonbhadra, S.K.; Agarwal, S.; Nagabhushan, P.; Syafrullah, M.; Adivarta, K. Privacy control in social networks by trust aware link prediction. In Proceedings of the 2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Bandung, Indonesia, 18–20 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 410–415.
12. Folino, F.; Pizzuti, C. Link prediction approaches for disease networks. In Proceedings of the International Conference on Information Technology in Bio-and Medical Informatics, Vienna, Austria, 4–5 September 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 99–108.
13. Gul, H.; Al-Obeidat, F.; Moreira, F.; Tahir, M.; Amin, A. Real-World Protein Particle Network Reconstruction Based on Advanced Hybrid Features. In Proceedings of the International Conference on Information Technology and Applications, Dubai, United Arab Emirates, 13–14 November 2021; Springer: Berlin/Heidelberg, Germany, 2022; pp. 15–22.
14. Lü, L.; Zhou, T. Linkprediction in complex networks: A survey. *Phys. A Stat. Mech. Appl.* **2011**, *390*, 1150–1170. [[CrossRef](#)]
15. Pavlov, M.; Ichise, R. Finding experts by link prediction in co-authorship networks. *FEWS* **2007**, *290*, 42–55.
16. Zhang, Q.M.; Shang, M.S.; Lü, L. Similarity-based classification in partially labeled networks. *Int. J. Mod. Phys. C* **2010**, *21*, 813–824. [[CrossRef](#)]
17. Lei, C.; Ruan, J. A novel link prediction algorithm for reconstructing protein–protein interaction networks by topological similarity. *Bioinformatics* **2013**, *29*, 355–364. [[CrossRef](#)] [[PubMed](#)]
18. Pinto, M.; Rodrigues, A.; Varajão, J.; Gonçalves, R. Model of functionalities for the development of B2B E-Commerce solutions. In *Innovations in SMEs and Conducting E-Business: Technologies, Trends and Solutions*; IGI Global: Hershey, PA, USA, 2011; pp. 35–60.
19. Martínez, V.; Berzal, F.; Cubero, J.C. A survey of link prediction in complex networks. *ACM Comput. Surv. (CSUR)* **2016**, *49*, 1–33. [[CrossRef](#)]
20. Aghabozorgi, F.; Khayyambashi, M.R. A new similarity measure for link prediction based on local structures in social networks. *Phys. A Stat. Mech. Appl.* **2018**, *501*, 12–23. [[CrossRef](#)]
21. Pecli, A.; Cavalcanti, M.C.; Goldschmidt, R. Automatic feature selection for supervised learning in link prediction applications: A comparative study. *Knowl. Inf. Syst.* **2018**, *56*, 85–121. [[CrossRef](#)]
22. Li, S.; Liu, M.; Li, H.; Hui, Y.; Chen, Z. Effects of structural damping on wind-induced responses of a 243-meter-high solar tower based on a novel elastic test model. *J. Wind Eng. Ind. Aerodyn.* **2018**, *172*, 1–11. [[CrossRef](#)]
23. Wasim, M. Link Prediction Using Double Degree Equation with Mutual and Popular Nodes. In *Trends and Applications in Information Systems and Technologies: Volume 4*; Springer: Berlin/Heidelberg, Germany, 2021; p. 328.
24. Gao, H.; Huang, J.; Cheng, Q.; Sun, H.; Wang, B.; Li, H. Link prediction based on linear dynamical response. *Phys. A Stat. Mech. Appl.* **2019**, *527*, 121397. [[CrossRef](#)]
25. Clauset, A.; Moore, C.; Newman, M.E. Hierarchical structure and the prediction of missing links in networks. *Nature* **2008**, *453*, 98. [[CrossRef](#)]
26. Zhu, B.; Xia, Y. An information-theoretic model for link prediction in complex networks. *Sci. Rep.* **2015**, *5*, 13707. [[CrossRef](#)]
27. Pech, R.; Hao, D.; Pan, L.; Cheng, H.; Zhou, T. Link prediction via matrix completion. *EPL (Europhys. Lett.)* **2017**, *117*, 38002. [[CrossRef](#)]
28. Prajapati, K.; Shah, H.; Mehta, R. A Survey of Link Prediction in Social Network Using Deep Learning Approach. 2020. Available online: <http://ir.paruluniversity.ac.in/xmlui/handle/123456789/7878> (accessed on 28 June 2022).
29. Xu, J.; Liu, A.; Xiong, N.; Wang, T.; Zuo, Z. Integrated collaborative filtering recommendation in social cyber-physical systems. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 1550147717749745. [[CrossRef](#)]
30. Esslimani, I.; Brun, A.; Boyer, A. Densifying a behavioral recommender system by social networks link prediction methods. *Soc. Netw. Anal. Min.* **2011**, *1*, 159–172. [[CrossRef](#)]
31. Perez-Cervantes, E.; Mena-Chalco, J.P.; De Oliveira, M.C.F.; Cesar, R.M. Using link prediction to estimate the collaborative influence of researchers. In Proceedings of the 2013 IEEE 9th International Conference on e-Science, Beijing, China, 22–25 October 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 293–300.
32. Huang, S.; Ma, L. Social Network Link Prediction Algorithm Based on Node Similarity. In Proceedings of the 2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), Dalian, China, 14–16 April 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1357–1360.
33. Lü, L.; Jin, C.H.; Zhou, T. Similarity index based on local paths for link prediction of complex networks. *Phys. Rev. E* **2009**, *80*, 046122. [[CrossRef](#)] [[PubMed](#)]
34. Liu, Y.; Liu, S.; Yu, F.; Yang, X. Link prediction algorithm based on the initial information contribution of nodes. *Inf. Sci.* **2022**, *608*, 1591–1616. [[CrossRef](#)]

35. Kerrache, S.; Alharbi, R.; Benhidour, H. A scalable similarity-popularity link prediction method. *Sci. Rep.* **2020**, *10*, 6394. [[CrossRef](#)]
36. Bai, M.; Hu, K.; Tang, Y. Link prediction based on a semi-local similarity index. *Chin. Phys. B* **2011**, *20*, 128902. [[CrossRef](#)]
37. Liu, W.; Lü, L. Link prediction based on local random walk. *EPL (Europhys. Lett.)* **2010**, *89*, 58007. [[CrossRef](#)]
38. Divakaran, A.; Mohan, A. Temporal link prediction: A survey. *New Gener. Comput.* **2020**, *38*, 213–258. [[CrossRef](#)]
39. Wang, H.; Le, Z. Seven-layer model in complex networks link prediction: A survey. *Sensors* **2020**, *20*, 6560. [[CrossRef](#)]
40. Wang, P.; Xu, B.; Wu, Y.; Zhou, X. Link prediction in social networks: The state-of-the-art. *Sci. China Inf. Sci.* **2015**, *58*, 1–38. [[CrossRef](#)]
41. Jaccard, P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bull. Soc. Vaudoise Sci. Nat.* **1901**, *37*, 547–579.
42. Adamic, L.A.; Adar, E. Friends and neighbors on the web. *Soc. Netw.* **2003**, *25*, 211–230. [[CrossRef](#)]
43. Zhou, T.; Lü, L.; Zhang, Y.C. Predicting missing links via local information. *Eur. Phys. J. B* **2009**, *71*, 623–630. [[CrossRef](#)]
44. Feng, X.; Zhao, J.; Xu, K. Link prediction in complex networks: A clustering perspective. *Eur. Phys. J. B* **2012**, *85*, 3. [[CrossRef](#)]
45. Barabási, A.L.; Jeong, H.; Néda, Z.; Ravasz, E.; Schubert, A.; Vicsek, T. Evolution of the social network of scientific collaborations. *Phys. A Stat. Mech. Appl.* **2002**, *311*, 590–614. [[CrossRef](#)]
46. Ravasz, E.; Somera, A.L.; Mongru, D.A.; Oltvai, Z.N.; Barabási, A.L. Hierarchical organization of modularity in metabolic networks. *Science* **2002**, *297*, 1551–1555. [[CrossRef](#)] [[PubMed](#)]
47. Leicht, E.A.; Holme, P.; Newman, M.E. Vertex similarity in networks. *Phys. Rev. E* **2006**, *73*, 026120. [[CrossRef](#)]
48. Zhu, Y.X.; Lü, L.; Zhang, Q.M.; Zhou, T. Uncovering missing links with cold ends. *Phys. A Stat. Mech. Appl.* **2012**, *391*, 5769–5778. [[CrossRef](#)]
49. Ng, A.; Jordan, M.; Weiss, Y. On spectral clustering: Analysis and an algorithm. *Adv. Neural Inf. Process. Syst.* **2001**, *14*, 849–856.
50. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. *Computer* **2009**, *42*, 30–37. [[CrossRef](#)]
51. Airoldi, E.M.; Blei, D.; Fienberg, S.; Xing, E. Mixed membership stochastic blockmodels. *Adv. Neural Inf. Process. Syst.* **2008**, *21*.
52. Perozzi, B.; Al-Rfou, R.; Skiena, S. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710.
53. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1067–1077.
54. Qiu, J.; Dong, Y.; Ma, H.; Li, J.; Wang, K.; Tang, J. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Los Angeles, CA, USA, 5–9 February 2018; pp. 459–467.
55. Kipf, T.N.; Welling, M. Variational graph auto-encoders. *arXiv* **2016**, arXiv:1611.07308.
56. Lichtenwalter, R.N.; Lussier, J.T.; Chawla, N.V. New perspectives and methods in link prediction. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–28 July 2010; pp. 243–252.
57. Findik, O.; Özkaynak, E. Link prediction based on node weighting in complex networks. *Soft Comput.* **2021**, *25*, 2467–2482. [[CrossRef](#)]
58. Jinseop, K.; Marcus, K. From *Caenorhabditis elegans* to the human connectome: A specific modular organization increases metabolic, functional and developmental efficiency. *Philos. Trans. R. Soc. B* **2014**, *369*, 20130529.
59. Colizza, V.; Pastor-Satorras, R.; Vespignani, A. Reaction-diffusion processes and metapopulation models in heterogeneous networks. *Nat. Phys.* **2007**, *3*, 027104. [[CrossRef](#)]
60. Rossi, R.A.; Ahmed, N.K. The Network Data Repository with Interactive Graph Analytics and Visualization. In Proceedings of the AAAI, Austin, TX, USA, 25–30 January 2015.
61. Spring, N.; Mahajan, R.; Wetherall, D. Measuring ISP topologies with Rocketfuel. In Proceedings of the SIGCOMM, Pittsburgh, PA, USA, 19–23 August 2002; Volume 32, pp. 133–145.
62. von Mering, C.; Krause, R.; Snel, B.; Cornell, M.; Oliver, S.G.; Fields, S.; Bork, P. Comparative assessment of large-scale data sets of protein–protein interactions. *Nature* **2002**, *417*, 399–403. [[CrossRef](#)]
63. Zachary, W.W. An Information Flow Model for Conflict and Fission in Small Groups. *J. Anthropol. Res.* **1977**, *33*, 452–473. [[CrossRef](#)]
64. Kunegis, J. Konect: The koblenz network collection. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; pp. 1343–1350.
65. Leskovec, J.; Kleinberg, J.; Faloutsos, C. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data (TKDD)* **2007**, *1*, 2-es. [[CrossRef](#)]
66. Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
67. Keikha, M.M.; Rahgozar, M.; Asadpour, M. Community aware random walk for network embedding. *Knowl.-Based Syst.* **2018**, *148*, 47–54. [[CrossRef](#)]
68. Pech, R.; Hao, D.; Lee, Y.L.; Yuan, Y.; Zhou, T. Link prediction via linear optimization. *Phys. A Stat. Mech. Appl.* **2019**, *528*, 121319. [[CrossRef](#)]
69. Yang, J.; Zhang, X.D. Predicting missing links in complex networks based on common neighbors and distance. *Sci. Rep.* **2016**, *6*, 38208. [[CrossRef](#)] [[PubMed](#)]