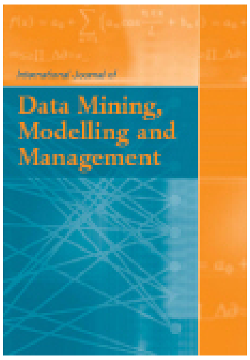


International Journal of Data Mining, Modelling and Management

ISSN online: 1759-1171 - ISSN print: 1759-1163
https://www.inderscience.com/ijdmmm

Enhancing link prediction in dynamic social networks: a novel algorithm integrating global and local topological structures

Shambhu Kumar, Arti Jain, Dinesh C.S. Bisht

DOI: 10.1504/IJDMMM.2025.10064902

Article History:

Received:	05 January 2024
Last revised:	18 April 2024
Accepted:	22 April 2024
Published online:	25 February 2025

Enhancing link prediction in dynamic social networks: a novel algorithm integrating global and local topological structures

Shambhu Kumar and Arti Jain

Department of Computer Science and
Engineering and Information Technology,
Jaypee Institute of Information Technology,
Noida, India
Email: kumar.shambhu@gmail.com
Email: ajain.jiit@gmail.com

Dinesh C.S. Bisht*

Department of Mathematics,
Jaypee Institute of Information Technology,
Noida, India
Email: drbisht.math@gmail.com
*Corresponding author

Abstract: The link prediction problem has gained significant importance due to the emergence of many social networks. Existing link prediction algorithms in social networks often prioritise local or global attributes, yielding satisfactory performance on specific network types but with limitations like reduced accuracy or higher computational burden. This paper presents a novel link prediction approach that integrates global and local topological structures, assessing node similarity through a similarity index formula between two node pairs that is based on three key features: the number of common neighbours between nodes with some penalty factor introduced for each common node, node influence, and the shortest path distance between unconnected nodes. Evaluation using AUC has been performed against seven datasets and demonstrates significant improvement over baseline and state-of-the-art methods, enhancing accuracy by 30% and 6.75%. This highlights the efficacy of integrating global and local features for more accurate link prediction.

Keywords: social network; link prediction; common neighbour; similarity measure; degree centrality; node distance.

Reference to this paper should be made as follows: Kumar, S., Jain, A. and Bisht, D.C.S. (2025) ‘Enhancing link prediction in dynamic social networks: a novel algorithm integrating global and local topological structures’, *Int. J. Data Mining, Modelling and Management*, Vol. 17, No. 1, pp.26–53.

Biographical notes: Shambhu Kumar is a research scholar in the Department of Computer Science and Engineering and Information Technology at Jaypee Institute of Information Technology (JIIT), Noida, Uttar Pradesh, India. He holds an undergraduate degree in Mathematics (2002) from Delhi University

and an MCA (2005) from the University of Hyderabad, Hyderabad, India. His research interests include machine learning, data science, data analytics, and social network analysis.

Arti Jain is a Senior Member of IEEE and an Assistant Professor (Sr. Grade) in the Computer Science and Engineering and IT Department at Jaypee Institute of Information Technology, Noida. With 21 years of academic and research experience, she is a member of IEEE, IAENG, INSTICC, and IFERP. She reviews for journals like Springer and IEEE, has guest-edited three special issues, and actively participates in international conferences. She has published four books and over 45 research papers. She supervises two PhD candidates, focusing on NLP, machine learning, data science, and related fields.

Dinesh C.S. Bisht is an Associate Professor of Mathematics at Jaypee Institute of Information Technology, Noida. He holds a PhD from G.B. Pant University and has over 14 years of experience. His research in soft computing, fuzzy optimization, fuzzy time series, and multicriteria decision making has resulted in over 50 publications in reputed international journals. He is a member of professional associations and an Associate Editor for the *International Journal of Mathematical, Engineering, and Management Sciences*, with awards for his reviewing contributions in the *Applied Soft Computing Journal*, Elsevier.

1 Introduction

Social networks are essential in today's world for educating individuals about current events, facilitating discussions, and connecting them to new people and organisations. It has significantly enhanced communication accessibility and has changed how people live nowadays. It is a modern way of exploring where people interact with others, make new friends, and become acquaintances with business customers. To do so, it undergoes the social network analysis (SNA) (Wasserman and Faust, 1994), which investigates social structure through usages of networks and graph theory. In the SNA, nodes are people or business entities, while associations, relationships and interactions among these entities are represented as edges. A link prediction problem in social networks is the likelihood of discovering future ties between nodes for which the current link is non-existent. Several recommendation systems (RS) can benefit, such as recommending new products on e-commerce sites, recommending new friends on social media sites, and recommending hotels through travel sites (Kaya, 2020). Also, there are a few more excellent examples of link prediction, such as co-authorship evolution prediction (Raeini, 2020), criminal network analysis (Berlusconi et al., 2016; Bedjou and Azouaou, 2023), the spread of epidemic disease (Palaniappan et al., 2022; Gupta and Gharehgozli, 2022), spam e-mail filtering (Huang et al., 2005), protein-to-protein interaction (Lei and Ruan, 2013), etc. Overall, link prediction serves as a fundamental tool for analysing social networks, understanding human behaviour, addressing various real-world challenges across different domains, and modelling social network data for processing (Aydin and Anderson, 2020). As social networks expand in complexity and size, it becomes increasingly crucial to develop sophisticated link prediction methods to extract valuable insights and optimise the utilisation of social network data. There are several practical implications of this research in real-world application across various fields/domains; a few of them are listed in Table 1.

Table 1 Practical implication of link prediction in different domains

Sr. no.	*Domain*	*Description*
1	Enhanced network analysis (Daud et al., 2020)	The research findings could lead to improved methods for predicting future connections in networks, such as social networks, communication networks, or biological networks.
2	Recommendation systems (Berkani, 2021)	The research findings could contribute to the development of more accurate recommendation systems in online platforms, such as social media, e-commerce websites, or content streaming services.
3	Fraud detection (Pourhabibi et al., 2020)	The research findings could be applied to detect fraudulent activities or anomalous behaviour in financial networks or online platforms.
4	Collaboration and innovation (Xi et al., 2024)	The research findings could identify the relationship between network properties to facilitate collaboration and innovation in research and development environments.
5	Resource allocation (Lei et al., 2022)	The research findings could optimise transportation networks in resource allocation, improve network efficiency, and mitigate potential bottlenecks or congestion by predicting future connections between nodes in the network.
6	Healthcare networks (Madani et al., 2023; Son and Kim, 2024)	The research findings could help in healthcare networks for potential collaborations between healthcare providers, researchers, or institutions.

The three major approaches for performing link predictions in the SNA are listed in Table 2.

Table 2 Major approaches link prediction

Sr. no.	*Approach*	*Description*
1	Similarity-based approach (Liben-Nowell and Kleinberg, 2007)	It is based on a graph's nodes' structural similarity, with two nodes with a high similarity index being the likelihood of joining subsequently. Nodes with a high similarity index are more likely to connect later.
2	Probabilistic approach (Wang et al., 2007)	It is focused on the characteristics (attributes) of edges and the behaviour of nodes.
3	Machine learning approach (de Sa and Prudencio, 2011)	It is based on a classification model developed on the network features.

Most researchers have primarily focused their research on a similarity-based approach due to low computing cost and higher efficiency. The link prediction method using a similarity-based approach is further subdivided into three categories, as stated in Table 3.

Existing similarity-based methods often concentrate on either local or global network attributes, demonstrating satisfactory performance on specific network types but also presenting drawbacks such as diminished prediction accuracy or heightened computational complexity. The global approaches are exceedingly time-consuming and expensive to compute, whereas the local approaches are less accurate than the global ones and less expensive to compute. However, the quasi-local approaches have lower processing costs and greater precision compared to the local approaches.

Table 3 Categorisation of similarity-based approach

Sr. no.	*Category*	*Description*	*Characteristics*	*Common technique*
1	Global approach (Liben-Nowell and Kleinberg, 2007)	The global approach uses the overall structure of a graph to determine how similar the nodes are to one another. This approach considers each set of nodes for a potential future link. Utilising the shortest distance between two nodes is the most used strategy.	• Better accuracy • Very high computational cost	Kartz index Leicht-Holme-Newman index (LHN) (Leicht et al., 2006)
2	Local approach (Liben-Nowell and Kleinberg, 2007)	The local approach uses only a small portion of the graph in order to obtain the local data necessary for prediction. This method considers a pair of nodes that fall into a selected portion of the graph. The most used technique is one based on the common neighbour between a pair of nodes.	• Lower accuracy • Very low computational cost • Faster and scalable	Common neighbour (CN) (Yang and Zhang, 2016) Jaccard index (JI) (Jaccard, 1901)
3	Quasi local (Wang et al., 2017)	The quasi-local approach uses a combination of global and local approaches. In addition to the indices used in the local approach, this method also makes use of indices used in the global approach.	• Accuracy > local approach • Computational cost > local approach and very lesser than global approach	

Over time, social networks have witnessed a proliferation of applications across various domains, leading to the emergence of numerous link prediction techniques in recent years. Many of these techniques focus on specific network topological attributes, with some emphasising either local or global attributes. Methods reliant solely on local topology, such as common neighbours (Yang and Zhang, 2016) and Jaccard coefficient (Jaccard, 1901), are computationally less intensive but tend to yield less accurate results. Conversely, in recent years, there has been a rise in the adoption of random walk-based learning approaches like node2vec (Grover and Leskovec, 2016), struc2vec (Figueiredo, 2017), and deep walk (Berahmand et al., 2021). Deep convolutional neural network-based approach (Wang et al., 2019), but the performance of these approaches typically falls short of in comparison with the topology-based similarity index.

Achieving optimal performance in link prediction involves navigating a delicate balance between accuracy and computational cost. While incorporating more features can enhance accuracy, it inevitably escalates computational overhead. Conversely, opting for fewer features may expedite computation but often yields less precise results. Thus, striking a balance by selecting an optimal set of features becomes imperative to achieve the desired level of accuracy without incurring excessive computational burden. Based on this survey, this research aims to devise a link prediction approach with higher prediction accuracy and a lower computational cost. This provides a new algorithm for link prediction attributed to network similarity while utilising the quasi-local technique with a view to achieving the same goal. The proposed algorithm identifies potential future links or relationships between nodes using three features, namely common neighbours with penalty, node-distance, and degree-centrality.

- Common neighbour with penalty (Rafiee et al., 2020): it refers to the shared collection of nodes that exist between neighbours of any two given nodes. The common neighbour plays a pivotal role in the link prediction, but its value gets penalised by the popularity of each common neighbour. The underlying assumption is that a very popular entity in a society may have many friends or followers in the network; hence, a more popular node in a common neighbour has lesser significance in determining attraction between the nodes.
- Node-distance (Kerrache et al., 2020): it makes reference to the shortest link between any two nodes, which also denotes their affinity. The affinity between nodes is inversely correlated with their separation.
- Degree-centrality (Li et al., 2018): it indicates how many edges connect a node. The highest degree centrality node is the most powerful node in the network and tends to draw other nodes to it.

Among the three parameters, common neighbour with penalty and degree-centrality are node-based local similarity indices (Liben-Nowell and Kleinberg, 2007), whereas node-distance is a path-based global similarity index (Liben-Nowell and Kleinberg, 2007). This paper assigns equal weightage to all these parameters, and their average derives the final similarity index between the nodes.

The proposed approach is evaluated on seven different datasets of varying node sizes, namely Zachary Karate Club (Zachary, 1977), Dolphins (Lusseau et al., 2003), US Airline (Xu and Harriss, 2008), Circuit (http://www.weizmann.ac.il/mcb/UriAlon/download/collection-complex-networks), E-mail (Guimerà et al., 2003), Football (Girvan and Newman, 2002), Power Grid (Us Power Grid Network Dataset-KONECT, 2016, n.d.; Watts and Strogatz, 1998). The approach is also compared with available benchmark algorithms for accuracy metrics. It is observed that the proposed approach gives the best result in comparison to all other algorithms with an overall area under the receiver operating characteristics curve (AUC) (Hanley and McNeil, 1982) of 97.2%.

1.1 Formal problem statement

In social networks, a graph $G(V, E)$ can be used to describe the link prediction problem where V is the representative of the collection of network nodes, vertices and users, whereas E is the representative of the set of network edges and relationships. The underlying assumption is that graph G is an undirected, unweighted, connected graph without self-loops and that there is only one edge between any two vertices.

Let

G undirected/unweighted graph with no self-loop

V set of nodes in G

E set of edges in G

n number of nodes in G

U set of all total possible edges in G

$|U|$ maximum number of total possible undirected edges that can occur in G.

Hence

$$n = |V| \tag{1}$$

$$|U| = |(n * (n-1))/2| \tag{2}$$

Also,

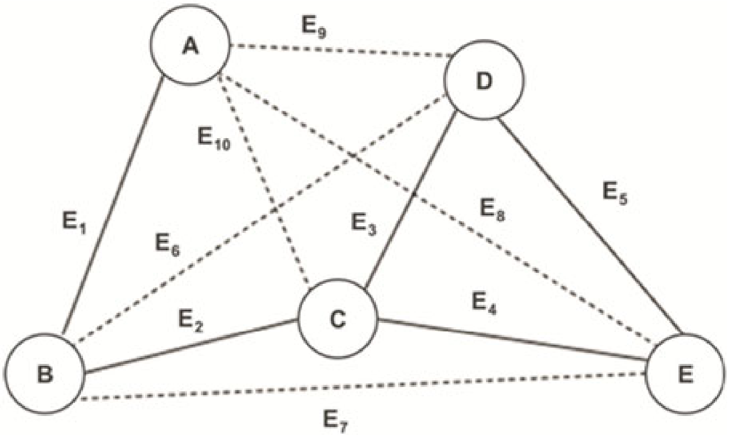
$U - E$ set of edges that are not in G but might come up later

$|U - E|$ number of edges that are absent in G but could be introduced later.

This link prediction problem seeks to anticipate which of the $U - E$ (non-existing edges) elements will appear in the forthcoming futuristic node. It can be achieved by calculating the similarity index value, i.e., $s(x, y)$ for each non-existing edge between node x and node y. A higher similarity index indicates that this edge will most likely come up in the near future. If $s(x, y) \geq a$ where 'a' is a positive threshold value; thereafter, a future connection between x and y will be predicted.

For example, consider an undirected network graph of 5 nodes, as represented in Figure 1.

Figure 1 Exemplified network



As shown in Figure 1,

$$V = \{A, B, C, D, E\}$$

$$n = |V| = 5$$

$$E = \{E_1, E_2, E_3, E_4, E_5\}$$

$$|E| = 5$$

$$U = \{E_1, E_2, E_3, E_4, E_5, E_6, E_7, E_8, E_9, E_{10}\}$$

$$|U| = (5*(5-1))/2 = 10$$

$$U - E = E_n = \{E_6, E_7, E_8, E_9, E_{10}\},$$

$U - E$: non-exsisting edges represented by dotted lines

$$|U - E| = 5.$$

The link prediction problem's suggested objective is to forecast future links or edges using an algorithm given a set of non-existing edges E_n.

2 Literature survey

A lot of research is being done on how to extract useful information from social networks, and link prediction is one of the most sophisticated research topics in this field. Jibouni et al. (2018) proposed a parameter-free method for link prediction based on a quasi-local similarity approach. It calculates the similarity index depending on each node's distance and degree. Their experiment has identified that prediction accuracy increases whenever the distance between nodes is either 2 or 3, and the AUC result outperforms other benchmark algorithms in a few of the data like Football (Girvan and Newman, 2002) and Power Grid (Us Power Grid Network Dataset-KONECT, October 2016, n.d.). Ahmad et al. (2020) have presented a link prediction algorithm that depends on common neighbours and closeness centrality between nodes. They have proposed a weighted relationship between common neighbours and closeness centrality between nodes. The node distance and the proximity centrality are inversely related. It is observed that 80% weightage to common neighbours and 20% weightage to closeness centrality achieve an accuracy of approximately 76%. However, finding a minimum distance between two nodes is time-consuming, and complexity increases when the graph becomes bigger. Zareie and Sakellariou (2020) have explained a link prediction algorithm based on common neighbours and second-order neighbours, also known as latent relationships. A similarity index is suggested utilising the Pearson relationship coefficient between the common second-order neighbours and common neighbours, but its accuracy couldn't exceed 80% in any of the datasets. The accuracy of the algorithm must be evaluated while considering the node's popularity. Kerrache et al. (2020) have suggested a global approach depending on popularity, the neighbour-based attraction between nodes, and path-based similarity between nodes. It is noted that a weight network matrix factors out popularity and attractiveness, specifically based on the network structure. Through Dijkstra's shortest path distance (Dijkstra, 1959; Frana and Misa, 2010), the weight matrix is used to calculate the dissimilarity between non-adjacent nodes. This approach takes the horizon cut-off distance as the two-path distance. It implies that if there is more than a two-hop distance between nodes, then the distance is taken as infinity. Through experimentation, it is proved that the algorithm produces highly accurate predictions with low time complexity in comparison with both the global and local methods. Yu et al. (2017) have published a similarity index based on a technique called path and node-based approach (PNC). The PNC approach improves the similarity-based link prediction compared to the individual path-based or node-based approach. This approach works well with both directed and undirected networks. Yang et al. (2018) have proved that endpoint influence represented by the node degree is not very influential in predicting the upcoming connections between two nodes. In contrast, it is proved that the influence is determined by the relations built through the path joining the in-between endpoints. It indicates that strong relationships are built through common neighbours with shorter paths, especially two-hop paths that bring more influence than the relations made through longer paths with weaker influence. Yuliansyah et al. (2023) introduced a method termed ‘degree of gravity for link prediction (DGLP)’ to tackle the issue of predicting links in a social network, whenever new nodes are introduced in future networks without available node attributes or network structure information. DGLP builds upon the triadic concept, which involves connections among friends of a friend, and it is influenced by Newton’s law of gravity. This approach integrates degree centrality, common neighbours, and the distance between nodes. Zhu et al. (2023) introduced a link prediction algorithm named DCCLP (stands for degree centrality of node pairs and the proximity centrality of nodes), which integrates both degree centrality and proximity centrality of nodes. The DCCLP algorithm underscores the importance of nodes in improving link prediction accuracy. However, the results of DCCLP indicate that this algorithm may not be ideal for large or sparse graphs. Sserwadda et al. (2023) propose an approach named topological similarity and centrality driven hybrid deep learning model for temporal link prediction (TSC-TLP) of processing the topological and centrality information of graph through deep learning model to get the graph embedding while preserving the topological information of graph. Further, embeddings improve the link prediction accuracy. Zhou et al. (2023) introduce a network embedding with nearest neighbours walk for a link prediction model to predict the link prediction. This approach identifies the nearest neighbour nodes using a natural nearest neighbour method. Then, it directs the walk based on the clustering coefficient of nodes to produce node sequences. Finally, these node sequences are inputted into a word2vec model to obtain node vectors utilised for link prediction. The link prediction technique is divided into normalised and un-normalised techniques. The normalised approach provides a similarity score between 0 and 1, whereas the un-normalised approach provides a similarity score more than 0. Azam et al. (2023) provide a comparison between the state-of-the-art normalised and un-normalised link prediction techniques and illustrate that normalised similarity provides better results than un-normalised one in link prediction. Aziz et al. (2023) presented a framework based on a parameterised matrix forest index (PMFI) and resource allocation (RA) index that is linked to a diffusion process on a network. The proposed method takes into account both the neighbourhood structure of two disconnected nodes and their relative positions within the network to predict the probability of a link forming between them. The network edges are assigned weights to facilitate quicker information flow between nodes with a high probability of future connection compared to those disconnected nodes less likely to be linked. This weighting is accomplished through the utilisation of the well-known resource allocation index. Gui (2024) introduced a methodology that regards the network’s structure as an intrinsic attribute and evaluates node similarity using non-trivial eigenvectors of the Laplacian matrix, employing metrics like Euclidean distance, Manhattan distance, and angular distance. Subsequently, classical machine learning algorithms can be applied for classification prediction, with a focus on two-class classification, facilitating the task of link prediction. Li et al. (2024) introduced a method that concentrates on enhancing the prediction accuracy of current local similarity-based techniques by incorporating local structural details and node degree information along 3-hop paths. Choudhury (2024) proposes a method for dynamic link prediction that considers evolving node relationships over time, extracting dynamic features from networks and employing machine learning techniques for future link prediction. Rai et al. (2023) suggested a similarity-based link prediction with the use of three network topology features: common neighbour, information transfer between two nodes through common neighbour and closeness between the nodes. Kumar et al. (2023) proposed that integrating both local and global topological features enhances prediction accuracy. The research survey highlights a significant challenge encountered by similarity-based algorithms: effectively identifying both local and global features of the graph, followed by selecting the appropriate features. While a greater number of features tend to enhance performance, it also escalates computational time. Conversely, a reduced number of features may expedite computational processes but often compromises algorithm performance. Hence, there arises a necessity for an algorithm that strikes an optimal balance between global and local features, ensuring high accuracy without exceeding computational time constraints. In the proposed algorithm, a similarity index formula is introduced based on three features, and its aim is to maintain time complexity comparable to baseline algorithms while enhancing accuracy.

3 Experimental setup

This section consists of Subsection 3.1 – dataset, Subsection 3.2 – evaluation strategies and Subsection 3.3 – proposed algorithms.

3.1 Dataset

In this research work, seven different real-world complex graph network datasets are taken care. These are publically available and very popular datasets. A brief description of each of these datasets is discussed here, along with their key attributes, nodes, edges, and non-edges, as described in Table 4. The Appendix is provided at the end of this manuscript, offering definitions for all acronyms used throughout the research paper, facilitating reader understanding and reference.

1 *Zachary Karate Club* (Zachary, 1977): it is a dataset containing the social relationships between 34 people of the university karate club.

2 *Dolphins* (Lusseau et al., 2003): it is a network of 62 bottlenose dolphins where a link represents frequent associations between dolphins.

3 *US Airline* (Xu and Harriss, 2008): it is a US airline transportation system connecting the USA to other countries through airlines.

4 *Circuit* (http://www.weizmann.ac.il/mcb/UriAlon/download/collection-complex-networks): it is a network of wires and electronic parts of a circuit. Here, the wire is considered an edge, and the electronics part is a graph node.

5 *E-mail* (Guimerà et al., 2003): it is an e-mail exchange network where users are considered nodes and e-mail exchanges as edges.

6 *Football* (Girvan and Newman, 2002): it is a collection of Division IA collegiate American football games played in the regular fall of 2000 that was put together by M. Girvan and M. Newman. The values of the nodes identify the conferences to which they belong.

7 *Power Grid* (Us Power Grid Network Dataset-KONECT, October 2016, n.d.; Watts and Strogatz, 1998): it is an undirected and unweighted network that depicts the American Western States Power Grid.

Before being used in the experiment, the datasets have been subjected to preprocessing. This involves the removal of duplicate edges between nodes and eliminating self-loops (edges with the same source and destination). Furthermore, only the largest connected component is preserved for analysis when networks are disconnected and comprised of multiple connected components.

Table 4 Chosen datasets and their key attributes

Dataset	*Nodes*	*Edges*	*Non-edges*	*Average degree*	*Clustering coefficient*	*Density*	*Degree assortativity*
Zachary Karate Club	34	78	483	4.59	0.57	0.13	-0.475
Dolphins	62	159	1732	5.13	0.25	0.08	-0.043
US Airline	332	2,126	52,820	12.81	0.63	0.03	-0.207
Circuit	512	819	129,997	3.199	0.30	0.006	-0.234
E-mail	1,133	5,452	635,827	9.62	0.22	0.008	0.078
Football	115	613	5,942	10.66	0.40	0.09	0.162
Power Grid	4,941	6,594	12,197,676	2.66	0.10	0.0005	0.003

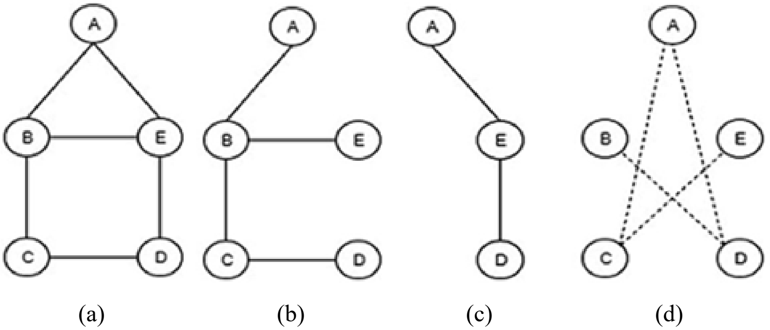
3.2 Evaluation strategy

Consider a graph dataset $G(V, E)$ that is split into the training and testing graphs, two separate, mutually incompatible graphs. A process randomly selects a few edges from the edges dataset E and forms the training graph G_t while the remaining edges, which are not part of the training graph, form the testing graph G_p. Edges that are part of the training graph (G_t) are named training edges (E_t) and edges that are part of the testing graph (G_p) are named testing edges (E_p). Here, E_t and E_p are mutually exclusive to each other and can be summed up as in equation (3).

$$E = E_t + E_p \tag{3}$$

Edges currently absent in graph G likely to appear in the future are represented by E_n. All these arrangements are depicted in Figure 2.

Figure 2 Graph G with training, testing and non-existing edges, (a) graph G with existing edges (E) (b) graph G_t with training edges (E_t) (c) graph G_p with testing edges (E_p) (d) graph G with non-existing edges (E_n)



In a series of rigorous experimentation, k-fold cross-validation (Stone, 1977) is applied, resulting in x% of edges from E are randomly selected to form the training graph G_t and remaining $(100 - x)$% edges from the testing graph G_p. The process is executed k number of times for each value of x, and an average of these k runs is marked as an outcome with x% of training data, and the overall average outcome is the average outcome marked with each value of x%. In this way, the outcome strengthens the applicability of the suggested algorithm. In this experiment, the value of k is 15, and the value of x varies from 50 to 90 (both values inclusive) with a step value of 5.

The hardware setup comprises an Intel(R) Xeon(R) Silver 4208 CPU @ 2.10GHz, with 64 GB of memory, operating on Red Hat Enterprise Linux Server release 7.9 (Maipo). The experiments are conducted entirely using Python version 3.6.8.

AUC is considered an evaluation criterion where AUC is calculated by selecting an edge from E_p (testing edges) and comparing its similarity score $s(x, y)$ against each non-existing edge from E_n using through equation (4). This process is repeated for each edge of E_p.

The AUC is computed as follows:

Let

n no. of independent comparison

n_1 no. of comparisons when there is a higher result for existing edges

n_2 no. of comparisons with the same score.

$$AUC = \frac{n_1 + 0.5n_2}{2} \tag{4}$$

In general, for a good link prediction, the AUC value should be nearer to 1. In the experimentation, since each edge from E_p gets compared against each edge from E_n, so n = number of edges in E_p * number of edges present in E_n; in other words,

$$n = |E_n| * |E_p| \tag{5}$$

3.3 Proposed algorithm

The proposed algorithm depends on three significant graph parameters: common neighbour with penalty, node-distance, and degree-centrality. The following sections go into great detail about these three parameters.

3.3.1 Common neighbour with penalty

The term ‘common neighbour’ describes how many nodes are shared by two given nodes. One takes into account each common node’s popularity and penalises the value of the common neighbour by the popularity of the common nodes. The penalty factor is derived from the inverse of the degree for each common neighbour node. This means that nodes with higher degrees contribute less to the common neighbour score, while nodes with lower degrees have a higher impact. Essentially, it penalises the contribution of highly connected nodes in calculating common neighbours, reflecting a preference for connections with the lesser popular nodes. The impact of the penalty factor on the algorithm’s performance depends on how it influences the resulting similarity scores between nodes. If the penalty factor is too harsh, it may disproportionately discount the influence of highly connected nodes, potentially leading to an underestimation of similarities. Conversely, if the penalty factor is too lenient, it may not effectively differentiate between common neighbours of varying popularity, affecting the accuracy of similarity assessments.

Mathematically, common neighbour with a penalty between node *x* and node *y* defined as *A*(*x*, *y*) and is stated in equation (6):

$$A(x, y)=\frac{\left[(|\urcorner(x)\cap\urcorner(y)|)-\sum_{t\in\urcorner(x)\cap\urcorner(y)}\frac{1}{k(t)}\right]}{\left|\urcorner(x)+\urcorner(y)-|\urcorner(x)\cap\urcorner(y)|\right|} \tag{6}$$

where

$\urcorner(x)$ set of neighbours of node *x*

$\urcorner(y)$ set of neighbours of node *y*

$k(t)$ degree of node *t*

n number of node in graph.

Theorem 1: *A*(*x*, *y*) is a normalised value that lies between 0 and 1.

Proof:

Case 1 (Worst case): When there is no common neighbour between node *x* and node *y*, then

$$|\urcorner(x)\cap\urcorner(y)|=0 \tag{7}$$

Hence, equation (6) can be written as

$$A(x, y)=0 \tag{8}$$

Case 2 (Best case): When all friends of node x are also friends of node y or vice-versa, or both node x and node y have the same set of friends, as in equation (9)

$$|\urcorner(x) \cap \urcorner(y)| = |\urcorner(x)| = |\urcorner(y)| \tag{9}$$

Hence, equation (6) can be written as

$$A(x, y) = \frac{\left[(|\urcorner(x)|) - \sum_{t \in \urcorner(x)} \frac{1}{k(t)} \right]}{|\urcorner(x)|} = 1 - \frac{\sum_{t \in \urcorner(x)} \frac{1}{k(t)}}{|\urcorner(x)|} \tag{10}$$

From equations (8) and (10), it is evident that $A(x, y)$ is a normalised value that lies between 0 and 1.

3.3.2 Node-distance

The shortest path distance quantifies the distance between two nodes regarding the number of edges that must be traversed to reach one node from the other. It provides insights into the network's connectivity and facilitates understanding of how easily information or influence can flow between nodes. Integrating shortest path distance into the similarity index allows us to consider direct connections and indirect paths between nodes, capturing the network's global structure and potential pathways for interactions. A shorter distance between nodes has a higher similarity, whereas a longer distance between nodes has a lesser similarity. This is known as node-distance. Mathematically, equation (11) states that the node-distance between nodes x and y is defined as $B(x, y)$.

$$B(x, y) = \frac{1}{1 + d(x, y)} \tag{11}$$

Here, $d(x, y)$: shortest distance between node x and node y.

To compute $d(x, y)$, Dijkstra's algorithm (Dijkstra, 1959; Frana and Misa, 2010) is considered to get the minimum distance between two nodes. If there is no path between two nodes, $d(x, y)$ tends to infinity. Also, it is observed that if two nodes are more than three-hop distance (Kerrache et al., 2020), then there is a rare chance that they will be a friend in the future; in such a case, the distance calculated between them tends to infinity.

Theorem 2: $B(x, y)$ is a normalised value that lies between 0 and 1.

Proof:

Case 1 (Worst case): When there does not exist any path between node x and node y, then $d(x, y) = \infty$ and so in equation (12).

$$B(x, y) = 0 \tag{12}$$

Case 2 (Best case): When node x and node y are just at one hop distance, then $d(x, y) = 1$, and so as in equation (13)

$$B(x, y) = 0.5 \tag{13}$$

From equations (12) and (13), it is evident that $B(x, y)$ is a normalised value that lies between 0 and 1.

3.3.3 Degree-centrality

The degree of a node in a graph is determined by the number of neighbouring connecting nodes, known as neighbour nodes. Nodes with a greater number of neighbours are deemed more popular within the network. This popularity is directly correlated with the node's degree, which measures its connectivity by counting the edges incident to it. Nodes exhibiting a higher degree centrality tend to assume more prominent roles in the network, potentially exerting greater influence. Integrating degree centrality into the similarity index allows for prioritising links between nodes with higher centrality, thereby capturing the network's structure and underscoring the significance of densely connected nodes. Degree-centrality between two nodes x and y is mathematically defined as $C(x, y)$ and is written as follows in equation (14):

$$C(x, y) = \frac{k(x) + k(y)}{2\ (maximum\ degree\ of\ \ graph)} \tag{14}$$

where

$k(x)$ degree of node x

$k(y)$ degree of node y.

Theorem 3: $C(x, y)$ is a normalised value that lies between 0 and 1.

Proof:

Case 1 (Worst case): The proposed algorithm deals with the connected network, so node x and node y have degree 1. In this case, $k(x) = k(y) = 1$, and equation (14) can be written as:

$$C(x, y) = \frac{1}{(maximum\ degree\ of\ \ a\ node\ in\ graph)} \tag{15}$$

So, $C(x, y)$ lies between 0 and 1.

Case 2 (Best case): Node x and node y share the network's highest degree. In the given case $k(x) = k(y) = m$ (maximum possible value) and so equation (14) can be written as:

$$C(x, y) = \frac{m + m}{2(m)} = 1 \tag{16}$$

So, $C(x, y)$ lies between 0 and 1.

From equations (15) and (16), it is evident that $C(x, y)$ is a normalised value that lies between 0 and 1.

3.3.4 Similarity formula

Combine equations (6), (11) and (14), the final similarity formula $S(x, y)$ between node x and node y is defined as an average of (x, y), $B(x, y)$ and $C(x, y)$ that can be rewritten as equation (17).

$$S(x, y) = (A(x, y) + B(x, y) + C(x, y))/3 \tag{17}$$

Theorem 4: $S(x, y)$ is a normalised value that lie between 0 and 1.

Proof: The value of $S(x, y)$ comprises of these parameters $A(x, y)$, $B(x, y)$ and $C(x, y)$. Each of these parameter values lies between 0 and 1 (Theorems 1–3).

Case 1: Worst case:

$$A(x, y) = B(x, y) = C(x, y) = 0 \tag{18}$$

Hence,

$$S(x, y) = 0 \tag{19}$$

Case 2: Best case:

$$A(x, y) = B(x, y) = C(x, y) = 1 \tag{20}$$

Hence,

$$S(x, y) = \frac{1+1+1}{3} = 1 \tag{21}$$

Thus, it is evident that $S(x, y)$ is a normalised value that falls between 0 and 1.

A greater similarity value of $S(x, y)$ means a higher chance of a future link and a lower similarity value of $S(x, y)$ means a lesser chance of future links. The algorithm of the proposed model is described in detail herewith.

3.3.5 Time complexity

- The time complexity for computing $A(x, y)$ for all possible pair of n nodes is $O(n \cdot k_{\max})$, where $k_{\max}$ represents the maximum degree of the graph. This complexity arises from traversing the entire graph to determine $k_{\max}$. As the process of computing $k_{\max}$ requires traversing the complete n nodes and n comparisons, the final time complexity for calculating $A(x, y)$ become $O(n^2)$.
- The time complexity of computing $B(x, y)$ is determined solely by the efficiency of Dijkstra's algorithm. For unweighted graphs, the time complexity of Dijkstra's algorithm is $O((n + m)\log n)$, where n represents the number of nodes and m represents the number of edges in the graph; however, if the input graph is sparse, then $|M| = O(N)$ and complexity become $O(n \log n)$.
- The time complexity to computing $C(x, y)$ is contingent on time required to traverse the graph and finding the node with maximum degree. Thus, the time complexity to get the maximum degree is $O(n^2)$, where n denotes the number of nodes in the graph.

The time complexity of $S(x, y)$ is determined by the sum of three components: $O(n^2)$, $O(n \log n)$ and $O(n^2)$, resulting in $O(n^2)$ overall.

Algorithm of the proposed model

INPUTS: Graph $G(V, E)$

OUTPUT: The likelihood of link existence between non-connected nodes through AUC value

```
Split $G$ into two graphs, i.e., $G_t$: training graph and $G_p$: testing graph respectively.
Search for $E_n$: non-existing edges in $G$.
Search for $E_p$: testing edges in $G$.
For each $x \in E_n$
    Calculate $S(x, y)$ using equation (17) and mark as $x_i$, where i = 1, 2 … $|E_n|$
End For
For each $y \in E_p$
    Calculate $S(x, y)$ using equation (17) and mark as $y_j$, where j = 1, 2 … $|E_p|$
End For
Compute $n = |E_n| * |E_p|$
AUC ← 0
For each $x_i$
    For each $y_j$
        Compute $n_1$: number of iterations, where $x_i > y_j$
        Calculate $n_2$: number of iterations, where $x_i = y_j$
    $x$ ← Compute $((n_1 + 0.5n_2)/n)$ using values $(n)$, $(n_1)$ and $(n_2)$
    AUC ← (AUC + $x$)
End For
Return AUC
```

To evaluate the suggested algorithm's predicted accuracy, its prediction accuracy (AUC value) was compared against eight different benchmark algorithms of link prediction. The eight different benchmark algorithms used in this manuscript are stated in Table 5.

Table 5 Baseline algorithms

$$\begin{pmatrix} \urcorner(x) \text{ and } \urcorner(y) \text{ neighbours of node } x \text{ and node } y, \text{ respectively} \\ Kx, Ky \text{ degrees of node } x \text{ and node } y, \text{ respectively.} \end{pmatrix}$$

Sr. No.	*Algorithm*	*Similarity formula*	*Time complexity*	*Remarks*
1	Common neighbour (CN) (Yang and Zhang, 2016)	$s(x, y) = \lvert \urcorner(x) \cap \urcorner(y) \rvert$	$O(n^2)$	If two nodes have a lot of common friends, it is most likely they become friend in future.
2	Preferential attachment index (PA) (Newman, 2001)	$s(x, y) = Kx \times Ky$	$O(n^2)$	If two nodes have an extensive connection in the network, they are most likely to link in the future.

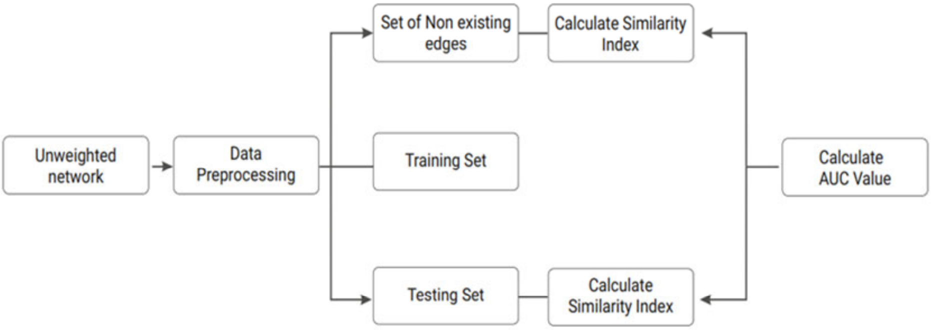
Table 5 Baseline algorithms (continued)

Sr. no.	*Algorithm*	*Similarity formula*	*Time complexity*	*Remarks*
3	Jaccard index (JI) (Lü et al., 2009)	$s(x, y) = \frac{\lvert \daleth(x) \cap \daleth(y) \rvert}{\lvert \daleth(x) \cup \daleth(y) \rvert}$	$O(n^2)$	It is the fraction of a common neighbour to all the neighbours between two nodes X and Y.
4	Hub promoted index (HPI) (Bliss et al., 2014)	$s(x, y) = \frac{\lvert \daleth(x) \cap \daleth(y) \rvert}{\min\{Kx, Ky\}}$	$O(n^2)$	It is the fraction of a common neighbour to the minimal degree of nodes X and Y.
5	Hub depressed index (HDI) (Bliss et al., 2014)	$s(x, y) = \frac{\lvert \daleth(x) \cap \daleth(y) \rvert}{\max\{Kx, Ky\}}$	$O(n^2)$	It is the fraction of a common neighbour to the maximal degree of nodes X and Y.
6	Sorensen index (SI) (Newman, 2001)	$s(x, y) = \frac{2\lvert \daleth(x) \cap \daleth(y) \rvert}{Kx + Ky}$	$O(n^2)$	It is the fraction of the common neighbour to the degree sum of the two nodes, X and Y.
7	Salton index (cosine similarity) (SAL) (Leydesdorff, 2008)	$s(x, y) = \frac{\lvert \daleth(x) \cap \daleth(y) \rvert}{\sqrt{Kx + Ky}}$	$O(n^2)$	It is the estimate of the cosine angle with respect to adjacency matrix of node X and Y.

3.4 Block diagram to visually represent the algorithm process

Figure 3 represents the block diagram of different processes for the proposed algorithm.

Figure 3 Average AUC value on different datasets



4 Results and discussion

4.1 Result comparison with benchmark algorithm.

Table 6 compiles the findings based on the AUC value achieved from different algorithms on various datasets of different sizes. The dataset also includes information on

AUC's standard deviation. This experiment is based on a k-fold cross-validation experiment in which the training graph G_t is created using x% of the edges from the input graph E and the testing graph G_p is created using the remaining (100 – x)% of the edges. In this experiment, the value of k is 15, and the value of x lies between 50 and 90 (both inclusive) with a step value of 5. Here, standard deviation indicates the deviation of the AUC value for each value of *x*for a given dataset. The proposed algorithm's standard deviation is smaller than that of existing algorithms, demonstrating that the distribution of the training and test datasets has no discernible impact on the AUC value.

Additionally, Table 6 demonstrates that the suggested approach performs better than the other benchmarked algorithms, and its results are consistently higher across the different datasets of varying sizes. The results show that the suggested algorithm has underperformed on the datasets for Karate and Dolphin. Still, its average accuracy is 47% better than the average benchmark AUC of the Karate dataset and 21 % better with the Dolphin network. This could be because the proposed algorithm's accuracy improves with the more extensive network, whereas Karate and Dolphin are relatively small networks. Another reason could be that Karate (Zachary, 1977) and Dolphin (Lusseau et al., 2003) networks are naturally occurring datasets with minimal human intervention.

Table 6 AUC and standard deviation of the algorithms on selected datasets

	Karate (KRT)	*Dolphin (DLN)*	*Football (FBT)*	*USAir (UAI)*	*Circuit (CKT)*	*E-mail (EML)*	*Power grid (PRG)*
CN	0.697 (0.027)	0.796 (0.016)	0.853 (0.009)	0.940 (0.001)	0.559 (0.006)	0.863 (0.003)	0.601 (0.002)
PA	0.856 (0.018)	0.746 (0.016)	0.538 (0.010)	0.901 (0.002)	0.668 (0.009)	0.804 (0.002)	0.723 (0.003)
JI	0.562(0.019)	0.780 (0.015)	0.855 (0.010)	0.901 (0.001)	0.559 (0.006)	0.859 (0.003)	0.601 (0.002)
HPI	0.622 (0.016)	0.755 (0.013)	0.854(0.010)	0.850 (0.001)	0.559 (0.006)	0.854 (0.003)	0.601 (0.002)
HDI	0.557 (0.019)	0.787 (0.016)	0.854 (0.010)	0.894 (0.002)	0.559 (0.006)	0.860 (0.003)	0.601 (0.002)
SI	0.562 (0.019)	0.780 (0.015)	0.855 (0.010)	0.901 (0.001)	0.559 (0.006)	0.859 (0.003)	0.601 (0.002)
SAL	0.569 (0.018)	0.771 (0.014)	0.855 (0.010)	0.909 (0.001)	0.559 (0.006)	0.858 (0.003)	0.601 (0.002)
LHN	0.512 (0.010)	0.747 (0.012)	0.854 (0.010)	0.739 (0.002)	0.559 (0.006)	0.849 (0.003)	0.601 (0.002)
Proposed	0.913 (0.009)	0.945 (0.006)	0.984 (0.001)	0.987 (0.0004)	0.992 (0.0001)	0.988 (.0002)	0.999 (8.18638E-06)

In contrast, the other five networks are human-designed to serve specific purposes. The findings suggest that naturally occurring networks demonstrate intricate, less deterministic connectivity patterns. Such networks are frequently influenced by stochastic processes, evolutionary dynamics, and external factors, resulting in heightened variability and unpredictability in link formation. Consequently, the process of link prediction in naturally occurring networks is anticipated to encounter more formidable challenges and exhibit lower accuracy levels than purpose-specific manmade networks.

The proposed method achieves the highest AUC value for the Power Grid dataset, which may be attributed to the network's small assortative coefficient. This characteristic leads to a scenario where most nodes exhibit similar degrees of connectivity. Similarly, the proposed approach yields favorable results for the football network, characterised by a small assortative coefficient, resulting in nodes with comparable degrees of connectivity.

Figure 4 Average AUC value of the different algorithm (see online version for colours)

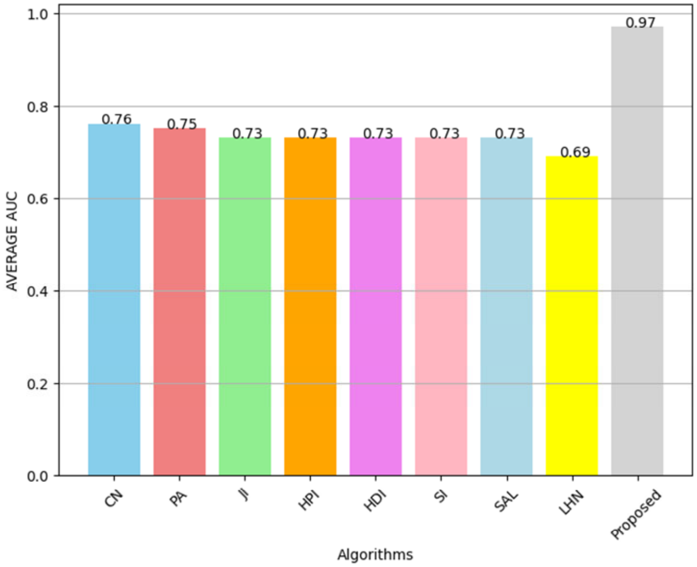
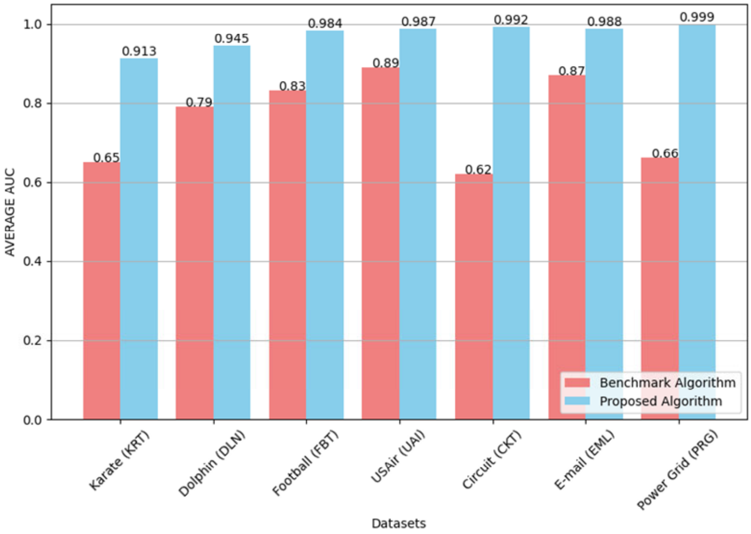


Figure 5 Average AUC value on different datasets (see online version for colours)



However, the outcomes demonstrate that the average AUC of the proposed method is 30% better than the average AUC of all other benchmark algorithms. Figure 4 also indicates that the proposed algorithm's average AUC is the highest compared to different benchmark algorithms.

Next, the algorithms' overall effectiveness on the chosen datasets of varying data sizes has been demonstrated. This is to find out which datasets are more complex to predict than others. Calculate the average AUC value across all chosen algorithms to obtain this information on every dataset. This information is summarised in Figure 5. It is visible from the figure that the benchmark algorithm performance is very suboptimal compared to the proposed algorithm.

The worst average AUC has been obtained in Circuit (0.62) and Karate network (0.65), whereas the highest average AUC has been achieved in USAir dataset (0.89). Even though predicting the future links is very difficult with these datasets, the proposed algorithm has outstanding accuracy with Circuit (0.992) and Karate (0.913).

By combining degree centrality and shortest path distance in the similarity index, this research aims to provide a more comprehensive measure of node similarity that accounts for both local and global network properties. This integrated approach offers a nuanced understanding of network dynamics, facilitating more accurate link predictions and revealing underlying patterns of connectivity and influence within the network.

In terms of time complexity, the proposed algorithm time complexity $O(n^2)$ is the same as the benchmark algorithm $O(n^2)$ however, it outperforms the average AUC of all other algorithms by 32%.

In summary, the proposed algorithm demonstrates consistently high accuracy across various dataset sizes and outperforms different benchmark algorithms while maintaining the same time complexity as the benchmarks.

4.2 Result comparison against state-of-the-art algorithms

In recent years, two prominent methods have emerged for predicting links in social networks: one utilises biased random walks, while the other employs deep graph convolutional neural networks (GCNs). Biased random walk techniques involve simulating random walks on the graph to capture local neighbourhood information, yielding low-dimensional node representations that preserve structural properties. On the other hand, deep GCNs analyse graph-structured data using convolutional layers to learn hierarchical node and edge representations.

In this research, the effectiveness of the new algorithm is assessed by comparing it with Node2Vec, which relies on random walks, and DenseNet-LP, which utilises deep graph convolutional networks. node2vec (Grover and Leskovec, 2016) was executed using default settings, with the embedding dimension set to 64. DenseNet-LP (Wang et al., 2019), on the other hand, was configured with a single convolutional layer comprising 32 filters sized 4 × 4, along with average pooling of size 2 × 2. Additionally, it incorporates a fully connected neural network with two hidden layers, each containing 128 neurons. The model is trained over 50 epochs.

Figure 6 indicates the AUC value of the proposed algorithm against each of the seven datasets, along with the AUC value of the state-of-the-art algorithm Node2Vec and DenseNet-LP algorithms. It clearly indicates that the combination of topological-based local and global features over shines these latest algorithms. Figure 7 presents an average

AUC of state-of-art algorithm and proposed algorithm across all datasets, and it is visible that the proposed algorithm performance is 7.8% better than node2Vec and 5.7% better than DenseNet-LP.

Figure 7 clearly shows that the average AUC of the proposed algorithm outperforms the latest state-of-the-art algorithm by a significant margin.

Figure 6 AUC comparison of the proposed algorithm with the latest state-of-the-art algorithm

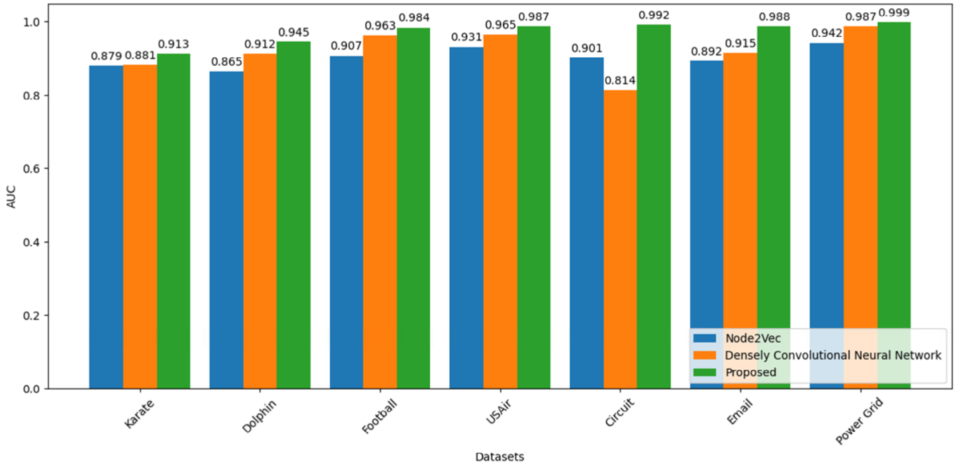
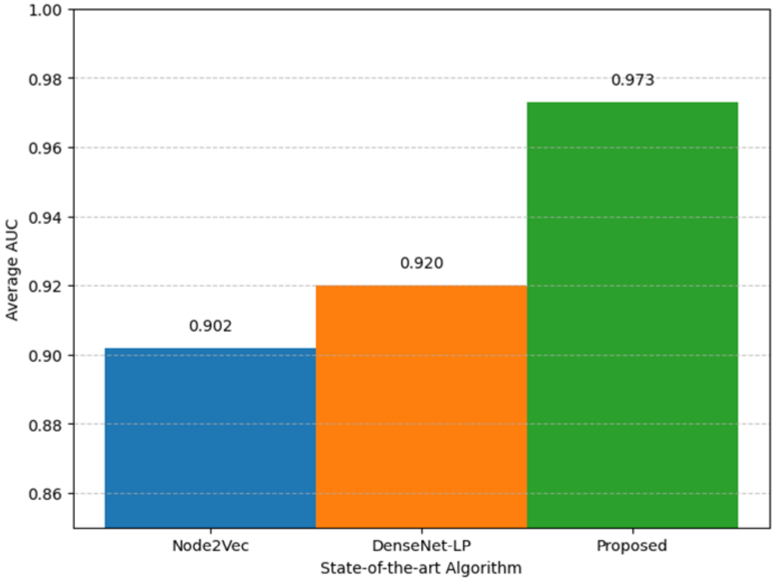


Figure 7 Comparison of average AUC value across dataset with state-of-art algorithms (see online version for colours)



4.3 Statistical analysis of the proposed algorithm

This section delves into a detailed discussion concerning the comprehensive statistical analysis of the AUC presented in Table 6, attained by the proposed approach, alongside comparisons to other benchmark algorithms outlined in Table 5. To assess the significance of differences between the proposed approach and existing benchmark algorithms, this study initially employs the Friedman test (Friedman, 1937) followed by the Holm (Haynes, 2013) procedure. The Friedman test ranks the observations within each group across all methods and compares the average ranks to ascertain statistical significance.

Table 7 displays the row rank table, which ranks the algorithms according to their AUC scores. It ranks the performance of each algorithm across multiple datasets, presents an overview of how each method compares in terms of its average rank and helps calculate the Friedman test statistics. Here, a higher rank indicates superior performance compared to other algorithms. Table 8 presents the parameters used for Q (Friedman statistics) and P-value generation. Under the null hypothesis, Q follows a chi-square distribution with k – 1 degrees of freedom; the corresponding p-value for Q = 27.27 is 0.00063, signifies rejection of the null hypothesis (0.00063 < 0.05), indicating significant differences among the compared methods and so now need to make next test (Holm procedure).

Table 7 Friedman row rank table

Dataset	*Proposed*	*CN*	*PA*	*JI*	*HPI*	*HDI*	*SI*	*SAL*	*LHN*
Karate (KRT)	9	7	8	3.5	6	2	3.5	5	1
Dolphin (DLN)	9	8	1	5.5	3	7	5.5	4	2
Football (FBT)	9	2	1	7	4	4	7	7	4
USAir (UAI)	9	8	5	5	2	3	5	7	1
Circuit (CKT)	9	4	8	4	4	4	4	4	4
E-mail (EML)	9	8	1	5.5	3	7	5.5	4	2
Power GRD (PRG)	9	4	8	4	4	4	4	4	4
Sum of ranks	63	41	32	34.5	26	31	34.5	35	18
Sum of ranks squared	3,969	1,681	1,024	1,190.25	676	961	1,190.25	1,225	324

Table 8 Parameters used for P-value and Q generation

k: number of algorithms	9
n: no. of dataset/observation	7
df: degree of freedom	8
Q (Friedman statistics)	27.27
p-value	0.0006

The computed Friedman statistics (Q = 27.27) is obtained using equation (22).

$$Q = \frac{12}{n \cdot k \cdot (k+1)} \sum\nolimits_{j=1}^{k} R_j^2 - 3 \cdot n \cdot (k+1) \tag{22}$$

Here, R_j is the sum of ranks for the j^{th} group.

Table 9 Pairwise comparisons of proposed algorithm vs. benchmark algorithm

Algorithms pair	*Holm-p value*	*Home inference*
Proposed vs. CN	0.0045310	$p < 0.01$
Proposed vs. PA	0.0060280	$p < 0.01$
Proposed vs. JI	0.0075799	$p < 0.01$
Proposed vs. HPI	0.0093184	$p < 0.01$
Proposed vs. HDI	0.0088318	$p < 0.01$
Proposed vs. SI	0.0060639	$p < 0.01$
Proposed vs. SAL	0.0046837	$p < 0.01$
Proposed vs. LHN	0.0025376	$p < 0.01$

Subsequently, Table 9 provides insights into pairwise comparisons between algorithms using the Holm method, indicating whether they are significantly different based on their average ranks and corresponding p-values. The 'Holm-p value' column shows the p-values obtained from these comparisons. In statistical hypothesis testing, if the p-value is less than a predetermined significance level (typically 0.05 or 0.01), it suggests that there is a statistically significant difference between the two compared algorithms. Here, the 'Holm inference' column states that the p-values for all comparisons are less than 0.01, indicating that there is a significant difference between the performance of the proposed algorithm and each of the benchmark algorithms at a 99% confidence level.

5 Conclusions

It is a challenging task to perform link prediction in a complex network. By addressing the challenges of link prediction in complex networks, the proposed algorithm offers a valuable tool for understanding and predicting future connections within dynamic systems. Real-world network sizes are enormous, so considering a complete network to find future links is time-consuming, compute-intensive, and complex. However, considering only local structure will miss a lot of network information that depreciates prediction accuracy. Considering these challenges, the approach in the proposed algorithm is quasi-local, where the combination of local topological structures and global factors significantly influences the prediction of future links. Its performance (AUC) was far superior to that of other benchmarked and state-of-art algorithms. It is evidentially proved that link prediction accuracy improved significantly after global (node distance) and local attributes (common neighbour with penalties and degree centrality) came together.

Future research avenues could encompass integrating edge weights into analysis frameworks to understand node connections' strengths and address directional network relationships, enhancing accuracy and applicability. Additionally, optimising weight assignment for components like a common neighbour with penalisation, node distance, and node degree using genetic algorithms or particle swarm optimisation could enhance link prediction significance. Researchers can further refine the penalty factor's parameter values through cross-validation or grid search methods to improve algorithm performance and link prediction accuracy. Moreover, the proposed algorithm's potential extends to addressing link prediction challenges in diverse network types, such as biological, communication, transportation, and financial networks, optimising operations and decision making. Exploring its effectiveness in dynamic networks could offer insights into adaptability and robustness across evolving network structures.

References

Ahmad, I., Akhtar, M.U., Noor, S. and Shahnaz, A. (2020) 'Missing link prediction using common neighbor and centrality based parameterized algorithm', *Scientific Reports*, Vol. 10, No. 1, p.364, https://doi.org/10.1038/s41598-019-57304-y.

Aydin, A.A. and Anderson, K.M. (2020) 'Data modelling for large-scale social media analytics: design challenges and lessons learned', *IJDMMM*, Vol. 12, No. 4, p.386, 2020, DOI: 10.1504/IJDMMM.2020.111409.

Azam, M., Nouman, M., Alfaouri, A.H., Saleh, A.M. and Abuaddous, H.Y. (2023) 'Evaluations of similarity base link prediction techniques in social network', *Journal of Engineering Science and Technology*, Vol. 18, No. 2, pp.1055–1082.

Aziz, F., Slater, L.T., Bravo-Merodio, L., Acharjee, A. and Gkoutos, G.V. (2023) 'Link prediction in complex network using information flow', *Sci. Rep.*, September, Vol. 13, No. 1, p.14660, DOI: 10.1038/s41598-023-41476-9.

Bedjou, K. and Azouaou, F. (2023) 'Detection of terrorism's apologies on Twitter using a new bi-lingual dataset', *IJDMMM*, Vol. 15, No. 4, pp.331–354, 2023, DOI: 10.1504/IJDMMM.2023.134581.

Berahmand, K., Nasiri, E., Rostami, M. and Forouzandeh, S. (2021) 'A modified DeepWalk method for link prediction in attributed social network', *Computing*, October, Vol. 103, No. 10, pp.2227–2249, DOI: 10.1007/s00607-021-00982-2.

Berkani, L. (2021) 'Recommendation of items using a social-based collaborative filtering approach and classification techniques', *IJDMMM*, Vol. 13, Nos. 1–2, p.137, DOI: 10.1504/IJDMMM.2021.112919.

Berlusconi, G., Calderoni, F., Parolini, N., Verani, M. and Piccardi, C. (2016) 'Link prediction in criminal networks: a tool for criminal intelligence analysis', *PLoS ONE*, April, Vol. 11, No. 4, p.e0154244, DOI: 10.1371/journal.pone.0154244.

Bliss, C.A., Frank, M.R., Danforth, C.M. and Dodds, P.S. (2014) 'An evolutionary algorithm approach to link prediction in dynamic social networks', *Journal of Computational Science*, Vol. 5, No. 5, pp.750–764, https://doi.org/10.1016/j.jocs.2014.01.003.

Choudhury, N. (2024) 'Community-aware evolution similarity for link prediction in dynamic social networks', *Mathematics*, January, Vol. 12, No. 2, p.285, DOI: 10.3390/math12020285.

Collection of Complex Networks [online] http://www.weizmann.ac.il/mcb/UriAlon/download/collection-complex-networks (accessed 29 September 2022).

Daud, N.N., Ab Hamid, S.H., Saadoon, M., Sahran, F. and Anuar, N.B. (2020) 'Applications of link prediction in social networks: a review', *Journal of Network and Computer Applications*, September, Vol. 166, p.102716, DOI: 10.1016/j.jnca.2020.102716.

de Sa, H.R. and Prudencio, R.B.C. (2011) 'Supervised link prediction in weighted networks', *The 2011 International Joint Conference on Neural Networks*, July, IEEE, San Jose, CA, USA, pp.2281–2288, DOI: 10.1109/IJCNN.2011.6033513.

Dijkstra, E.W. (1959) 'A note on two problems in connexion with graphs', *Numer. Math.*, December, Vol. 1, No. 1, pp.269–271, DOI: 10.1007/BF01386390.

Figueiredo, D.R. (2017) 'struc2vec: learning node representations from structural identity', *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August, pp.385–394, DOI: 10.1145/3097983.3098061.

Frana, P.L. and Misa, T.J. (2010) 'An interview with Edsger W. Dijkstra', *Commun. ACM*, August, Vol. 53, No. 8, pp.41–47, DOI: 10.1145/1787234.1787249.

Friedman, M. (1937) 'The use of ranks to avoid the assumption of normality implicit in the analysis of variance', *Journal of the American Statistical Association*, December, Vol. 32, No. 200, pp.675–701, DOI: 10.1080/01621459.1937.10503522.

Girvan, M. and Newman, M.E.J. (2002) 'Community structure in social and biological networks', *Proc. Natl. Acad. Sci. U.S.A.*, June, Vol. 99, No. 12, pp.7821–7826, DOI: 10.1073/pnas.122653799.

Grover, A. and Leskovec, J. (2016) 'node2vec: scalable feature learning for networks', *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August, ACM, San Francisco, California, USA, pp.855–864, DOI: 10.1145/2939672.2939754.

Gui, C. (2024) 'Link prediction based on spectral analysis', *PLoS ONE*, February, Vol. 19, No. 2, p.e0298926, DOI: https://doi.org/10.1371/journal.pone.0287385.

Guimerà, R., Danon, L., Díaz-Guilera, A., Giralt, F. and Arenas, A. (2003) 'Self-similar community structure in a network of human interactions', *Phys. Rev. E*, December, Vol. 68, No. 6, p.065103, DOI: 10.1103/PhysRevE.68.065103.

Gupta, A. and Gharehgozli, A. (2022) 'Developing a machine learning framework to determine the spread of COVID-19 in the USA using meteorological, social, and demographic factors', *IJDMMM*, Vol. 14, No. 2, p.89, DOI: 10.1504/IJDMMM.2022.123360.

Hanley, J.A. and McNeil, B.J. (1982) 'The meaning and use of the area under a receiver operating characteristic (ROC) curve', *Radiology*, April, Vol. 143, No. 1, pp.29–36, DOI: 10.1148/radiology.143.1.7063747.

Haynes, W. (2013) 'Holm's method', in Dubitzky, W., Wolkenhauer, O., Cho, K-H. and Yokota, H. (Eds.): *Encyclopedia of Systems Biology*, pp.902–902, Springer New York, New York, NY, DOI: 10.1007/978-1-4419-9863-7_1214.

Huang, Z., Li, X. and Chen, H. (2005) 'Link prediction approach to collaborative filtering', *Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries – JCDL '05* , ACM Press, Denver, CO, USA, p.141, DOI: 10.1145/1065385.1065415.

Jaccard, P. (1901) *Étude comparative de la distribution florale dans une portion des Alpes et du Jura*, DOI: 10.5169/SEALS-266450.

Jibouni, A., Lotfi, D., Marraki, M.E. and Hammouch, A. (2018) 'A novel parameter free approach for link prediction', *2018 6th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pp.1–6, https://doi.org/10.1109/WINCOM.2018.8629586.

Kaya, B. (2020) 'A hotel recommendation system based on customer location: a link prediction approach', *Multimed. Tools Appl.*, January, Vol. 79, No. 3–4, pp.1745–1758, DOI: 10.1007/s11042-019-08270-0.

Kerrache, S., Alharbi, R. and Benhidour, H. (2020) 'A scalable similarity-popularity link prediction method', *Sci. Rep.*, December, Vol. 10, No. 1, p.6394, DOI: 10.1038/s41598-020-62636-1.

KONECT (2016) *US Power Grid Network Dataset*, October.

Kumar, S., Jain, A. and Bisht, D. (2023) 'Hybrid approach for link prediction using supervised machine learning in social networks: combining global and local features', *Proceedings of the 2023 Fifteenth International Conference on Contemporary Computing*, August, ACM, Noida, India, pp.591–597, DOI: 10.1145/3607947.3608065.

Lei, C. and Ruan, J. (2013) 'A novel link prediction algorithm for reconstructing protein-protein interaction networks by topological similarity', *Bioinformatics*, February, Vol. 29, No. 3, pp.355–364, DOI: 10.1093/bioinformatics/bts688.

Lei, W., Alves, L.G.A. and Amaral, L.A.N. (2022) 'Forecasting the evolution of fast-changing transportation networks using machine learning', *Nat. Commun.*, July, Vol. 13, No. 1, p.4252, DOI: 10.1038/s41467-022-31911-2.

Leicht, E.A., Holme, P. and Newman, M.E.J. (2006) 'Vertex similarity in networks', *Phys. Rev. E*, February, Vol. 73, No. 2, p.026120, DOI: 10.1103/PhysRevE.73.026120.

Leydesdorff, L. (2008) 'On the normalization and visualization of author co-citation data: Salton's cosine versus the Jaccard index', *J. Am. Soc. Inf. Sci.*, January, Vol. 59, No. 1, pp.77–85, DOI: 10.1002/asi.20732.

Li, L., Liu, X., Chen, N. and Tian, H. (2018) 'Link prediction based on node centrality', *Proceedings of the International Conference on Information Technology and Electrical Engineering 2018*, December, ACM, Xiamen Fujian China, pp.1–6, DOI: 10.1145/3148453.3306256.

Li, T. et al. (2024) 'Link prediction based on local structure and node information along local paths', *The Computer Journal*, January, Vol. 67, No. 1, pp.45–56, DOI: 10.1093/comjnl/bxac157.

Liben-Nowell, D. and Kleinberg, J. (2007) 'The link-prediction problem for social networks', *J. Am. Soc. Inf. Sci.*, May, Vol. 58, No. 7, pp.1019–1031, DOI: 10.1002/asi.20591.

Lü, L., Jin, C-H. and Zhou, T. (2009) 'Similarity index based on local paths for link prediction of complex networks', *Phys. Rev. E*, October, Vol. 80, No. 4, p.046122, DOI: 10.1103/PhysRevE.80.046122.

Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E. and Dawson, S.M. (2003) 'The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations', *Behavioral Ecology and Sociobiology*, September, Vol. 54, No. 4, pp.396–405, DOI: 10.1007/s00265-003-0651-y.

Madani, A. et al. (2023) 'An ABC approach for depression signs on social networks posts', *IJDMMM*, Vol. 15, No. 3, pp.275–296, DOI: 10.1504/IJDMMM.2023.132972.

Newman, M.E.J. (2001) 'Clustering and preferential attachment in growing networks', *Phys. Rev. E*, July, Vol. 64, No. 2, p.025102, DOI: 10.1103/PhysRevE.64.025102.

Palaniappan, S., Ragavi, V., David, B. and Nisha, S.P. (2022) 'Prediction of epidemic disease dynamics on the infection risk using machine learning algorithms', *SN Comput. Sci.*, January, Vol. 3, No. 1, p.47, Jan. 2022, DOI: 10.1007/s42979-021-00902-3.

Pourhabibi, T., Ong, L.K., Kam, B.H. and Boo, Y.L. (2020) 'Fraud detection: a systematic literature review of graph-based anomaly detection approaches', *Decision Support Systems*, June, Vol. 133, p.113303, DOI: 10.1016/j.dss.2020.113303.

Raeini, M.G. (2020) *Link Prediction Using Supervised Machine Learning based on Aggregated and Topological Features*, arXiv:2006.16327[cs], June [online] http://arxiv.org/abs/2006.16327 (accessed 1 April 2022).

Rafiee, S., Salavati, C. and Abdollahpouri, A. (2020) 'CNDP: link prediction based on common neighbors degree penalization', *Physica A: Statistical Mechanics and its Applications*, February, Vol. 539, p.122950, DOI: 10.1016/j.physa.2019.122950.

Rai, A.K., Tripathi, S.P. and Yadav, R.K. (2023) 'A novel similarity-based parameterized method for link prediction', *Chaos, Solitons & Fractals*, October, Vol. 175, p.114046, DOI: 10.1016/j.chaos.2023.114046.

Son, J. and Kim, D. (2024) 'Applying network link prediction in drug discovery: an overview of the literature', *Expert Opinion on Drug Discovery*, January, Vol. 19, No. 1, pp.43–56, DOI: 10.1080/17460441.2023.2267020.

Sserwadda, A., Ozcan, A. and Yaslan, Y. (2023) 'Topological similarity and centrality driven hybrid deep learning for temporal link prediction', *JUCS*, May, Vol. 29, No. 5, pp.470–490, DOI: 10.3897/jucs.99169.

Stone, M. (1977) 'An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion', *Journal of the Royal Statistical Society: Series B (Methodological)*, September, Vol. 39, No. 1, pp.44–47, DOI: 10.1111/j.2517-6161.1977.tb01603.x.

Wang, C., Satuluri, V. and Parthasarathy, S. (2007) 'Local probabilistic models for link prediction', *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pp.322–331, https://doi.org/10.1109/ICDM.2007.108.

Wang, J., Ma, Y., Liu, M., Yuan, H., Shen, W. and Li, L. (2017) 'A vertex similarity index using community information to improve link prediction accuracy', *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, October, IEEE, Banff, AB, pp.158–163, DOI: 10.1109/SMC.2017.8122595.

Wang, W., Wu, L., Huang, Y., Wang, H. and Zhu, R. (2019) 'Link prediction based on deep convolutional neural network', *Information*, May, Vol. 10, No. 5, p.172, DOI: 10.3390/info10050172.

Wasserman, S. and Faust, K. (1994) *Social Network Analysis: Methods and Application*, Cambridge University Press.

Watts, D.J. and Strogatz, S.H. (1998) 'Collective dynamics of 'small-world' networks', *Nature*, June, Vol. 393, No. 6684, pp.440–442, DOI: 10.1038/30918.

Xi, X., Zhao, J., Yu, L. and Wang, C. (2024) 'Exploring the potentials of artificial intelligence towards carbon neutrality: Technological convergence forecasting through link prediction and community detection', *Computers & Industrial Engineering*, April, Vol. 190, p.110015, DOI: 10.1016/j.cie.2024.110015.

Xu, Z. and Harriss, R. (2008) 'Exploring the structure of the U.S. intercity passenger air transportation network: a weighted complex network approach', *GeoJournal*, July, Vol. 73, No. 2, p.87, DOI: 10.1007/s10708-008-9173-5.

Yang, J. and Zhang, X.D. (2016) 'Predicting missing links in complex networks based on common neighbors and distance', *Scientific Reports*, Vol. 6, No. 1, p.38208, https://doi.org/10.1038/srep38208.

Yang, Y., Zhang, J., Zhu, X. and Tian, L. (2018) 'Link prediction via significant influence', *Physica A: Statistical Mechanics and Its Applications*, Vol. 492, pp.1523–1530, https://doi.org/10.1016/j.physa.2017.11.078.

Yu, C., Zhao, X., An, L. and Lin, X. (2017) 'Similarity-based link prediction in social networks: a path and node combined approach', *Journal of Information Science*, Vol. 43, No. 5, pp.683–695, DOI: 10.1177/0165551516664039.

Yuliansyah, H., Othman, Z.A. and Bakar, A.A. (2023) 'A new link prediction method to alleviate the cold-start problem based on extending common neighbor and degree centrality', *Physica A: Statistical Mechanics and its Applications*, April, Vol. 616, p.128546, DOI: 10.1016/j.physa.2023.128546.

Zachary, W.W. (1977) 'An information flow model for conflict and fission in small groups', *Journal of Anthropological Research*, December, Vol. 33, No. 4, pp.452–473, DOI: 10.1086/jar.33.4.3629752.

Zareie, A. and Sakellariou, R. (2020) 'Similarity-based link prediction in social networks using latent relationships between the users', *Sci. Rep.*, December, Vol. 10, No. 1, p.20137, DOI: 10.1038/s41598-020-76799-4.

Zhou, M., Han, Q., Li, M., Li, K. and Qian, Z. (2023) 'Nearest neighbor walk network embedding for link prediction in complex networks', *Physica A: Statistical Mechanics and its Applications*, June, Vol. 620, p.128757, DOI: 10.1016/j.physa.2023.128757.

Zhu, J., Dai, F., Zhao, F. and Guo, W. (2023) 'Integrating node importance and network topological properties for link prediction in complex network', *Symmetry*, July, Vol. 15, No. 8, p.1492, DOI: 10.3390/sym15081492.

Appendix

Acronym	*Full form*
CKT	Circuit network dataset
CN	Common neighbour algorithm
DLN	Dolphins network dataset
EML	Email network dataset
FBT	Football dataset
HDI	Hub depressed index algorithm
HPI	Hub promoted index algorithm
JI	Jaccard index algorithm
KRT	Zachary karate club network
LHN	Leicht Holme Neman algorithm
OPF	Our proposed model
PA	Preferential attachment index algorithm
PRG	Power grid network dataset
RA	Random forest algorithm
SAL	Salton algorithm
SI	Soremsen algorithm
UAI	US airlines dataset