Bulletin of Electrical Engineering and Informatics
Vol. 13, No. 3, June 2024, pp. 1980∼1990
ISSN: 2302-9285, DOI: 10.11591/eei.v13i3.6761

A novel particle swarm optimization-based intelligence link prediction algorithm in real world networks

Deepjyoti Choudhury, Tapodhir Acharjee
Department of Computer Science and Engineering, Assam University, Silchar, India

Article Info

Article history:

Received May 19, 2023
Revised Oct 31, 2023
Accepted Dec 10, 2023

Keywords:

Common neighbour
Link prediction
Relationship
Social networks
Swarm

ABSTRACT

Link prediction in social network is an important topic due to its applications like finding collaborations and recommending friends. Among existing link prediction methods, similarity-based approaches are found to be most effective since they examine the number of common neighbours (CN). Current work presents a novel link prediction algorithm based on particle swarm optimization (PSO) and implemented on four real world datasets namely, Zachary's karate club (ZKC), bottlenose dolphin network (BDN), college football network (CFN) and Krebs' books on American politics (KBAP). It consists of three experiments: i) to find the measures on existing methods and compare them with our proposed algorithm; ii) to find the measured values of the existing methods along with our proposed one to determine future links among nodes that have no CN; and iii) to find the measures of the methods to determine future links among nodes having same number of CN. In experiment 1, our proposed approach achieved 75.88%, 78.34%, 82.63% and 78.36% accuracy for ZKC, BDN, CFN, and KBAP respectively. These results beat the performances of traditional algorithms. In experiment 2, the accuracies are found as 75.53%, 74.25%, 81.63% and 78.34% respectively. In experiment 3, accuracies are detected as 72.75%, 81.53%, 78.35% and 75.13% respectively.

Corresponding Author:

Deepjyoti Choudhury
Department of Computer Science and Engineering, Assam University
Silchar, India
Email: deepjyotichoudhury05@gmail.com

1. INTRODUCTION

It is very common that predicting future links in online social networks is a challenging task due to the vast number of users and huge number of connections. Collaborative networks are well explained by Jin *et al.* [1], Barabási *et al.* [2], and Davidsen *et al.* [3] that have been usually assessed by certain overall structural characteristics within networks. Connection prediction in this context makes it possible to infer that certain people are collaborating even when their interactions have not been directly observed [4]. The problems with link prediction are also linked to the issue of finding missing network connections [5], [6].

Liben-Nowell and Kleinberg [7] experimented on large co-authorship networks which provided information about future connections. Then, Lü and Zhou [8] summarised the then-recent progress of link prediction algorithms. Symeonidis *et al.* [9] defined a basic node similarity measure that captures effectively local graph features. Some sample link prediction approaches were examined by classifying them with model type [10]. Almansoori *et al.* [11] also proposed a link prediction model which was capable of predicting both links that might exist and disappear in the future. In the same year, Liu *et al.* [12] suggested an approach for link prediction which focused on social media data in order to identify the most influenced ones. Dong *et al.* [13] proposed two improved algorithms: CNGF algorithm based on local information and KatzGF algorithm based on a global information network. Bliss *et al.* [14] proposed a method for predicting future linkages by optimising weights. Tang *et al.* [15] presented a logical technique to utilise positive connections and content-centric user interactions.

Again, the problem of attacking similarity-based link prediction was presented once more [16]. Moraes *et al.* [17] presented a common approach to link prediction that computed degrees of compatibility between unconnected node pairs in the network. Mallek *et al.* [18] addressed the link prediction issue within the uncertain framework of belief function theory. In the same year, a systematic analysis of existing link prediction methodologies was provided [19]. Yuliansyah *et al.* [20] showed a systematic review and introduced the link prediction taxonomy. Kou *et al.* [21] presented that the existing link prediction research did not consider the need for user privacy protection. A line graph neural networks was proposed to predict links in social networks [22]. To experiment link prediction technique in dynamic networks, graph convolution network embedded long short term memory network (GC-LSTM), an end-to-end model long short term memory network (LSTM) was proposed by Chen *et al.* [23]. Utilizing the relationship between phenotype and genotype, a machine learning based link prediction algorithm was also proposed [24].

In social networks, the prediction of links indicate new or dissolution links in future networks [25]. Even if two nodes in a network don't share a common first-order neighbor, they could nevertheless be connected through a large number of intermediate neighbors. In this case, the relationship between two nodes is made clear by the number of their shared neighbors. This form of connection is referred to as a concealed relationship in this paper. We have generated two problem statements based on concealed relationship as follows:

– Statement 1: in real-world networks, there may be a substantial fraction of linkages between nodes that have no common neighbours (CN). Methods that rely solely on common first-order neighbours for prediction may have limited accuracy depending on the network.
– Statement 2: there is a large overlap in the same number of neighbours while potential links are developed among nodes after sorting all the existing links.

From the two statements we can generate two postulates:

– Postulate 1: connection creation can be expected if the nodes linked to a future link have no CN but have a substantial hidden association.
– Postulate 2: when hidden connections are taken into account, variations in existing and non-existing linkages between pairs of nodes with the same number of shared neighbours may be explained.

The swarm intelligence-based approaches have gained a lot of popularity in the various field because of their ability to converge to near optima and have low computational complexity [26]. Particle swarm optimization (PSO) [27] is a heuristic search approach whose artificers are inspired by biological populations' swarming or collaborative activity. Several problems have been solved using swarm intelligence based techniques and presented in the current year [28]–[31]. A comprehensive survey of whale optimization algorithm was presented by Mahmood *et al.* [32] recently. Applications of PSO have been observed in different fields very recently [33]–[36]. Jawad *et al.* [37] presented a modified meerkat clan algorithm using swarm intelligence technique. The same approach was presented by Saleh *et al.* [38] in different way. Truong *et al.* [39] succeeded in developing X-ware using intelligence based method that they tested against Windows-based PCs. Abdulhussein *et al.* [40] created a reliable cascade P-PI controller to regulate the speed and position of the permanent magnet DC motor. Hamed *et al.* [41] suggested a cooperative hunting approach for a multi-robot system based on the wolf swarm algorithm whose behaviour is unforeseen by numerous robots. Methods were also presented that combined universal, problem-free swarm intelligence algorithms with simple deterministic domain-specific heuristic algorithms [42], [43]. Application of swarm based intelligence technique can also be seen in image processing problems [44]. The swarm based approach was also applied to check robotic behaviour which could be scaled to industrial and consumer markets to further the ability of automation [45].

Based on postulate 1 and postulate 2, there are three different experiments conducted to evaluate our proposed approach: i) to find the results of the methods using different evaluative parameters; ii) to find the possible future links that have no CN but concealed relationships; and iii) to calculate the measures to determine future links among nodes that have the same number of CN. We have arranged this paper as follows: i) method is discussed in the section 2; ii) it is followed by results and discussion in section 3; and iii) finally our manuscript ends with an elaborated conclusion in section 4.

2. METHOD

The approaches for predicting hidden connections differ mostly in how they calculate the similarity score between two nodes, which is subsequently used to calculate the likelihood of each non-existing link. We have used eight common methods to predict concealed relationships in this paper: CN [46], preferential attachment (PA) [47], Jaccard coefficient (JC) [7], hub promoted index (HPI) [14], resource allocation (RA) [48], Adamic-Adar (AA) [49], Salton index (SI) [50], and Leicht-Holme-Newman (LHN) index. Figure 1 depicts the work flow of the model in which we have applied PSO to predict future links. In our proposed approach, PSO is used as the predictive model where Pearson correlation is considered as the fitness function. In this approach, we adapt the modified velocity and position update equation to solve the link prediction problem [51]. The distance between the particle and its Pbest and Gbest is calculated using the shortest path distance which in turn is used for calculating the velocity.

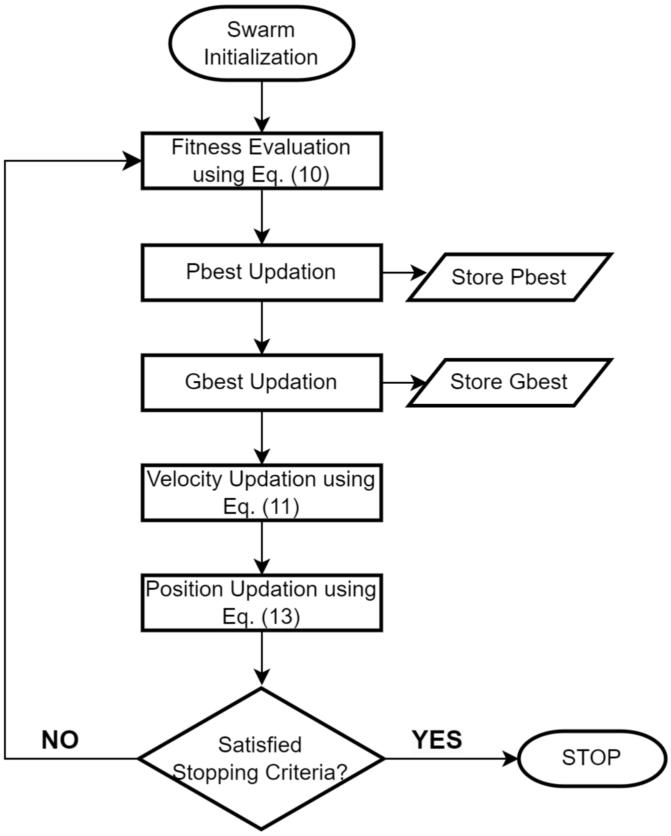

Figure 1. Work flow of the proposed method

The particle which has the best fitness and if it is better than the previous one, the particle is regarded as global leader. Then a random node is selected from the region to update the particle position. The details of each step is explained in the subsections mentioned in below.

2.1. Particle representation and fitness evaluation

Each particle of the swarm S has N dimensions where N represents the number of nodes in the graph. The i^{th} index of the particle and the j^{th} value at the i^{th} index represents the edge between the i^{th} and j^{th} nodes in the graph.

In order to calculate the likelihood of link formation between nodes v_i and v_j, the union neighbourhood set, U_{ij} is calculated as (1):

$$U_{ij} = \{z : (N_i[z] > 0) or (N_j[z] > 0)\} \tag{1}$$

Greater correlation between the union neighbourhood set, U_{ij}, of the vectors N_i and N_j indicates higher structural similarity between nodes i and j. To determine the correlation between two nodes, the correlation coefficient between the union neighbourhood set of vectors is determined. We use Pearson correlation coefficient to calculate the fitness of each particle. The correlation between the union neighbourhood set of the vectors N_i and N_j is calculated as (2):

$$CR_{ij} = \frac{\sum_{z \in U_{ij}} (N_i[z] - \overline{\mathrm{N_i}})(N_j[z] - \overline{\mathrm{N_j}})}{\sqrt{\sum_{z \in U_{ij}} (N_i[z] - \overline{\mathrm{N_i}})^2 (N_j[z] - \overline{\mathrm{N_j}})^2}} \tag{2}$$

$\overline{\mathrm{N_i}}$ is the mean of the values in the union neighbourhood set of vector N_i and it is calculated as: $\overline{\mathrm{N_i}} = \frac{\sum_{z \in U_{ij}} (N_i[z])}{U_{ij}}$. Even if two nodes have no shared neighbours in our approach, they may have considerable structural similarities. As a result, a link may be identified by comparing their neighbours.

Algorithm 1 Proposed PSO based Link Prediction Algorithm

Input: Read nodes from the real world networks
Output: Possible near future links among the nodes
1: Initialize Swarm Randomly
2: Fitness Evaluation of Each Particle based on Pearson Correlation
3: Find Pbest and Gbest
4: **while** terminating condition **do**
5:   **for** each particle **do**
6:   Calculate Velocity
7:   Update Position
8:   Fitness Evaluation
9:   Update Pbest
10:   **end for**
11:   Update Gbest
12:   **if** $TerminationCriteria = Reached$ **then**
13:     Stop & Execute
14:   **else if** $TerminationCriteria \neq Reached$ **then**
15:     Go to STEP 5
16:   **end if**
17: **end while**
18: End Algorithm

2.2. Velocity and position updation

We have modified the velocity and position updation for each particle. We update the velocity of a particle as (3):

$$\begin{aligned} {v_i}^{t+1} = w \otimes v^t + c_1 \otimes rand() \otimes dist(p_i, x_i) \\ + c_2 \otimes rand() \otimes dist(G_i, x_i) \end{aligned} \tag{3}$$

where, the symbol $\otimes$ denotes point by point vector multiplication. ${v_i}^{t+1}$ is the velocity of the i_{th} particle at (t + 1) time. The inertia momentum factor w; $(0 < w < 1)$; self-confidence factor c_1 and swarm confidence factor c_2 are non-negative real constants. We have taken the value of c_1 and c_2 are [1:4 - 1:9]. $dist(p_i, x_i)$ means the shortest path distance from the Pbest particles to the swarm particles. Similarly, $dist(G_i, x_i)$ denotes the shortest path distance from the Gbest particle to the swarm particles.

The position of the particle is not updated directly adding the velocity to each position. Rather we consider the velocity as the radius within which the particle can move. We find set T of such points as (4):

$$T = \{x_j : dist(x_i, x_j) \leq {v_i}^{t+1}\} \tag{4}$$

Now, we select randomly one such element from the set T to update the particle position as (5):

$$x_i = rand(T) \tag{5}$$

Now, the position for the particle is updated and if a better position is found in next iteration then the new position will be updated accordingly.

3. RESULTS AND DISCUSSION

3.1. Datasets

The proposed method is implemented on different real world datasets with satisfactory results. The names of the used datasets are: Zachary's karate club (ZKC) [52], bottlenose dolphin network (BDN) [53], college football network (CFN) [54], and Krebs' books on American politics (KBAP) [55]. The characteristics of the dataset is represented in a tabular format in Table 1.

Table 1. Characteristics of the datasets

SI no.	Dataset	Nodes	Edges	Average degree
1	ZKC	34	78	2.29
2	BDN	62	159	2.56
3	CFN	115	615	5.35
4	KBAP	105	441	4.20

3.2. Results analysis

Our experiments are implemented using Python programming language in Jupyter notebook, the system specification is Windows 10 OS with 8 GB RAM. We have considered four datasets mentioned for the experiments in our proposed method. The collection of existing edges, E, for each network is randomly partitioned into two sets: training edges set, E_{Train} and test edges set, E_{Test}. We choose δ percentage of edges at random as E_{Train} and the rest $(1 - \delta)$ percentage of edges as E_{Test}. The procedure is repeated 15 times to raise the confidence in the acquired results, and the average of the obtained results is provided in each experiment. We have considered here mainly five matrices to evaluate our algorithm as mentioned below: precision, recall, F1, accuracy, and area under curve (AUC). The AUC is calculated by computing the similarity score between the pair of nodes connected to each of the edges using an edge from E_{Test} and an edge from the set of non-existing edges, E_{NE}. The AUC is obtained when this method is done k times using (6):

$$AUC = \frac{k_1 + \frac{1}{2}k_2}{k} \tag{6}$$

In (6), the number of times the similarity score of the nodes connected by the edge chosen from the set E_{Test} exceeds the similarity score of the nodes connected by the edge chosen from the set E_{NE} is considered as k_1, and the number of times the two similarity scores are identical is called k_2. The AUC value ranges from [0, 1], with a larger number indicating more accuracy.

- Experiment 1: in the first experiment, we pick a number of δ as 80 since it is a value that has been utilised in prior relevant studies [56]. Precision, recall, F1, accuracy, and AUC values are then calculated for each of the nine approaches including our proposed method, and for each of the four networks. Tables 2-6 summarise the findings of precision, recall, F1, accurracy and AUC score consecutively. Table 2 displays the values of precision we have found in our experiment. In CFN network, CN and PA method have attained the highest precision with 77% and 69% among existing methods. Similarly, JC method achieved the highest precision in ZKC network with 76%. HPI and AA method gained the highest precision in KBAP network with 72% and 73%; and SI method gained the highest precision in BDN network with more than 73% among existing methods. Table 3 exhibits the recall values of all the methods we have perceived. Almost all the existing methods have found the highest recall values in CFN network. The obtained F1 score is shown in Table 4. Here also, all the existing methods have gained the highest F1 score except JC method. Tables 5 and 6 present the accuracy and AUC score of all existing methods along with our proposed method. All other approaches are outperformed by our proposed method. Among all the methods, our proposed approach has achieved the highest precision at 84.27%, recall value at 88.34%, F1 score at 86.26%, accuracy at 82.63%, and AUC value as 80.35%

in CFN. Also, our proposed approach is able to perform better than the existing algorithms in all other datasets.

Table 2. Characteristics of the datasets

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7362	0.6291	0.7641	0.7193	0.7180	0.6892	0.7191	0.6924	0.7831
2	BDN	0.7523	0.6834	0.7484	0.6920	0.7251	0.6924	0.7352	0.7183	0.7792
3	CFN	0.7705	0.6981	0.7391	0.7231	0.7620	0.7192	0.7285	0.7237	0.8427
4	KBAP	0.7285	0.6682	0.7510	0.7296	0.7182	0.7310	0.7097	0.7173	0.8139

Table 3. Recall values comparison with existing methods and proposed method

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7475	0.6835	0.7846	0.7729	0.8182	0.6958	0.7384	0.6847	0.8131
2	BDN	0.7723	0.7081	0.7823	0.7194	0.8139	0.6942	0.7185	0.7124	0.8404
3	CFN	0.8159	0.7294	0.7938	0.8185	0.8409	0.7263	0.7834	0.7397	0.8834
4	KBAP	0.7638	0.7162	0.7924	0.7325	0.8354	0.7047	0.7463	0.7048	0.8191

Table 4. F1 score comparison with existing methods and proposed method

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7418	0.6552	0.7742	0.7451	0.7648	0.6925	0.7286	0.6885	0.7978
2	BDN	0.7622	0.6955	0.7650	0.7054	0.7669	0.6933	0.7268	0.7153	0.8086
3	CFN	0.7926	0.7134	0.7655	0.7678	0.7995	0.7227	0.7550	0.7316	0.8626
4	KBAP	0.7457	0.6914	0.7711	0.7310	0.7724	0.7176	0.7275	0.7110	0.8165

Table 5. Accuracy comparison with existing methods and proposed method

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6936	0.6830	0.7385	0.7317	0.6628	0.6583	0.7174	0.6952	0.7588
2	BDN	0.7184	0.6739	0.7295	0.7285	0.6927	0.6398	0.7648	0.7048	0.7834
3	CFN	0.7371	0.7104	0.7749	0.7738	0.7319	0.6852	0.7528	0.7784	0.8263
4	KBAP	0.7046	0.6950	0.7428	0.7183	0.7286	0.6741	0.7397	0.7642	0.7931

Table 6. AUC score comparison with existing methods and proposed method

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6746	0.7056	0.6842	0.6511	0.6834	0.6575	0.7174	0.6812	0.7275
2	BDN	0.7382	0.6181	0.7378	0.6275	0.6925	0.6389	0.7038	0.6934	0.7526
3	CFN	0.7514	0.7852	0.7637	0.6815	0.6676	0.6824	0.6841	0.6727	0.8035
4	KBAP	0.7248	0.6903	0.7294	0.6923	0.6764	0.7032	0.6951	0.6834	0.7836

– Experiment 2: the purpose of this experiment is to test postulate 1, which concerns the approaches' capacity to discern linkages between nodes with no shared neighbours. We take the set of test edges E_{Test}, and the set of non-existing edges E_{NE} for each of the four networks. We choose the edges that connect nodes that have no CN and have a degree larger than one from these two groups. Then, for our proposed technique and all other approaches, we calculate the similarity score for each of these edges. Precision, recall, F1, accuracy, and AUC scores of the methods are shown in Tables 7-11. Table 7 represents that our proposed approach attained the highest precision value on CFN network followed by ZKC and KBAP. In the same network, the best recall value and F1 score in Table 8 have been obtained by our proposed method. Table 9 explains about F1 score for all the methods out of which our proposed has outperformed others. Table 10 represents that our suggested technique is more accurate than previous methods in distinguishing links between nodes with no shared neighbours. This experiment demonstrates that assessing the correlation between neighbourhood vectors is a useful way to find the indirect similarity among nodes when they don't have any shared neighbours. Finally, we represent the AUC score in Table 11 where our proposed method dominates the values comparing with the existing methods.

Table 7. Precision values of the methods to determine future links among nodes that have no CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6936	0.7184	0.6942	0.6641	0.6832	0.6142	0.7239	0.6521	0.7485
2	BDN	0.6829	0.7041	0.7195	0.6741	0.7042	0.6531	0.7174	0.6354	0.7259
3	CFN	0.7163	0.7457	0.7326	0.6818	0.7132	0.6684	0.7484	0.7173	0.7895
4	KBAP	0.6731	0.6925	0.7196	0.6264	0.6842	0.6951	0.7264	0.7043	0.7415

Table 8. Recall values of the methods to determine future links among nodes that have no CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7452	0.7504	0.7143	0.6952	0.7587	0.6835	0.7843	0.7835	0.8036
2	BDN	0.7239	0.7848	0.7643	0.6823	0.8145	0.6942	0.7723	0.7255	0.7935
3	CFN	0.7783	0.8192	0.7853	0.6674	0.8264	0.7032	0.8342	0.7426	0.8636
4	KBAP	0.7483	0.8257	0.7385	0.6253	0.7964	0.6935	0.8185	0.7175	0.8367

Table 9. F1 scores of the methods to determine future links among nodes that have no CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7185	0.7341	0.7041	0.6793	0.7190	0.6470	0.7529	0.7118	0.7751
2	BDN	0.7028	0.7423	0.7412	0.6782	0.7553	0.6730	0.7438	0.6775	0.7582
3	CFN	0.7460	0.7807	0.7580	0.6745	0.7656	0.6854	0.7890	0.7297	0.8249
4	KBAP	0.7087	0.7533	0.7289	0.6258	0.7360	0.6943	0.7697	0.7108	0.7862

Table 10. Accuracy of the methods to determine future links among nodes that have no CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7372	0.7529	0.6856	0.6873	0.7034	0.6893	0.7523	0.6934	0.7553
2	BDN	0.7281	0.7193	0.6942	0.6793	0.6973	0.6743	0.7334	0.7045	0.7425
3	CFN	0.7461	0.7728	0.7153	0.6954	0.7327	0.6935	0.7734	0.7523	0.8163
4	KBAP	0.7382	0.6924	0.7043	0.6364	0.7056	0.6734	0.7175	0.7454	0.7834

Table 11. AUC score of the methods to determine future links among nodes that have no CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.7246	0.7756	0.6412	0.6911	0.7025	0.6741	0.7017	0.6841	0.7875
2	BDN	0.7123	0.6795	0.7253	0.6526	0.6845	0.6642	0.7328	0.7172	0.7216
3	CFN	0.7135	0.7329	0.7748	0.7485	0.7292	0.6942	0.7641	0.7682	0.7859
4	KBAP	0.7417	0.6625	0.7362	0.7432	0.7193	0.6815	0.7218	0.7362	0.7561

– Experiment 3: the purpose of this experiment is to test postulate 2, which concerns the approaches' capacity to detect linkages between nodes with the same number of shared neighbours. To do so, we take the set of test edges E_{Test}, and the set of non-existing edges E_{NE}, for each of the four networks. We choose the edges that connect nodes with the same number of shared neighbours from these two sets. Then, using our proposed technique, we calculate the similarity score for each of these edges. Precision, recall, F1, accuracy, and the AUC scores of the methods are shown in Tables 12-16. In Table 12, CN and JC methods have gained the highest precision values in BDN network with 67% and 69%; LHN method has attained the highest in KBAP network; SI method in ZKC network and the rest of the methods have achieved the highest precision score in CFN network among the existing methods. Table 13 represents the recall values of all the methods. Almost all the existing methods have dominant recall values in CFN network. CN and JC methods have gained the highest F1 score in BDN network among existing methods as shown in Table 14. Rest of the existing methods have presiding F1 score in CFN network. While we find the accuracy, our proposed method provides the best value among all methods with more than 81% as displayed in Table 15. Table 16 displays the AUC score for all the existing methods comparing with our proposed method. Our proposed approach has outcomes that are near to the best approach in the ZKC and Krebs' book network. In the rest of the two networks, our proposed method outperformed the others. Overall, the findings of this experiment show that utilising a neighbourhood vector for nodes to analyse correlation is a reliable technique to discriminate between the test and non-existing edges of nodes with an equal number of CN.

Table 12. Precision values to determine future links among nodes having same number of CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6426	0.6375	0.6364	0.7175	0.6946	0.7011	0.7286	0.6935	0.7414
2	BDN	0.6735	0.6647	0.6946	0.7137	0.7198	0.7162	0.7164	0.6375	0.7264
3	CFN	0.6375	0.6847	0.6645	0.7396	0.7267	0.7256	0.7185	0.6474	0.7729
4	KBAP	0.6646	0.6456	0.6746	0.7285	0.6934	0.7191	0.6936	0.6723	0.7273

Table 13. Recall values to determine future links among nodes having same number of CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6845	0.6735	0.6835	0.7424	0.7025	0.7175	0.7532	0.7264	0.7836
2	BDN	0.7142	0.7243	0.7232	0.7253	0.7284	0.7436	0.7364	0.7454	0.7646
3	CFN	0.7462	0.7324	0.6845	0.7538	0.7385	0.7746	0.7465	0.7392	0.8183
4	KBAP	0.6935	0.7045	0.7153	0.7572	0.7046	0.7646	0.7274	0.7537	0.7836

Table 14. F1 score to determine future links among nodes having same number of CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6629	0.6550	0.6591	0.7297	0.6985	0.7092	0.7407	0.7096	0.7619
2	BDN	0.6933	0.6932	0.7086	0.7195	0.7241	0.7296	0.7263	0.6872	0.7450
3	CFN	0.6876	0.7077	0.6744	0.7466	0.7326	0.7493	0.7322	0.6903	0.7950
4	KBAP	0.6787	0.6738	0.6944	0.7426	0.6990	0.7412	0.7101	0.7107	0.7544

Table 15. Accuracy values to determine future links among nodes having same number of CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6265	0.5834	0.6275	0.7325	0.6935	0.6835	0.7185	0.6835	0.7275
2	BDN	0.6925	0.5924	0.6848	0.7593	0.7426	0.7045	0.7426	0.6635	0.8153
3	CFN	0.6745	0.6163	0.6924	0.7627	0.7285	0.7254	0.7745	0.7364	0.7835
4	KBAP	0.6835	0.6034	0.6756	0.7475	0.7026	0.7364	0.7494	0.7258	0.7513

Table 16. AUC score to determine future links among nodes having same number of CN

Sl no.	Network name	CN	PA	JC	HPI	RA	AA	SI	LHN	Proposed method
1	ZKC	0.6429	0.6395	0.6395	0.7113	0.6863	0.6734	0.7036	0.6627	0.6927
2	BDN	0.7374	0.7291	0.7291	0.7427	0.7167	0.6945	0.7374	0.6384	0.7728
3	CFN	0.6218	0.7862	0.7862	0.7531	0.7253	0.6924	0.7395	0.7035	0.7639
4	KBAP	0.7195	0.6748	0.6748	0.7361	0.6945	0.6718	0.7185	0.6924	0.7251

4. CONCLUSION

In social network analysis, the prediction of future linkages and the discovery of missing links have sparked a lot of interest. It has been approached in different ways, out of which many are based on the number of shared neighbours. The premise of this research is that the number of CN does not convey latent interactions between nodes. As a consequence, a unique PSO-based technique was presented to account for such latent associations, and its accuracy was compared to that of existing relevant methods, yielding higher accuracy results. Here, we summarized briefly the advancement of connection prediction research and highlighted the latest contributions in the field of link prediction. The important input to the connection forecast of the research of complex networks is the thorough knowledge of structural variables that influence algorithmic efficiency. This paper proposes a new particle swarm intelligence-based algorithm for link prediction. The results showed good performance of the proposed approach compared with existing algorithms as shown in section. However, the algorithm requires tuning of parameters for better performance of the algorithm. Thus we plan to make the algorithm adaptive for automatically adjusting the parameters.

REFERENCES

[1] E. M. Jin, M. Girvan, and M. E. J. Newman, "Structure of growing social networks," *Physical Review E*, vol. 64, no. 4, p. 046132, Sep. 2001, doi: 10.1103/PhysRevE.64.046132.

[2] A. L. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. P. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," *Physica A: Statistical Mechanics and its Applications*, vol. 311, no. 3–4, pp. 590–614, Aug. 2002, doi: 10.1016/S0378-4371(02)00736-7.

[3] J. Davidsen, H. Ebel, and S. Bornholdt, "Emergence of a small world from local interactions: modeling acquaintance networks," *Physical Review Letters*, vol. 88, no. 12, p. 128701, Mar. 2002, doi: 10.1103/PhysRevLett.88.128701.

[4] V. E. Krebs, "Mapping networks of terrorist cells," *Connections*, vol. 24, no. 3, pp. 43–52, 2002.

[5] D. S. Goldberg and F. P. Roth, "Assessing experimentally derived interactions in a small world," *Proceedings of the National Academy of Sciences*, vol. 100, no. 8, pp. 4372–4376, Apr. 2003, doi: 10.1073/pnas.0735871100.

[6] F. Liu, B. Liu, C. Sun, M. Liu, and X. Wang, "Deep belief network-based approaches for link prediction in signed social networks," *Entropy*, vol. 17, no. 4, pp. 2140–2169, Apr. 2015, doi: 10.3390/e17042140.

[7] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, New York, NY, USA: ACM, Nov. 2003, pp. 556–559. doi: 10.1145/956863.956972.

[8] L. Lü and T. Zhou, "Link prediction in complex networks: a survey," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150–1170, Mar. 2011, doi: 10.1016/j.physa.2010.11.027.

[9] P. Symeonidis, E. Tiakas, and Y. Manolopoulos, "Transitive node similarity for link prediction in social networks with positive and negative links," in *Proceedings of the Fourth ACM Conference on Recommender Systems*, New York, NY, USA: ACM, Sep. 2010, pp. 183–190. doi: 10.1145/1864708.1864744.

[10] M. Al Hasan and M. J. Zaki, "A survey of link prediction in social networks," in *Social Network Data Analytics*, Boston, MA: Springer US, 2011, pp. 243–275. doi: 10.1007/978-1-4419-8462-3_9.

[11] W. Almansoori *et al.*, "Link prediction and classification in social networks and its application in healthcare and systems biology," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 1, no. 1–2, pp. 27–36, Jun. 2012, doi: 10.1007/s13721-012-0005-7.

[12] F. Liu, B. Liu, X. Wang, M. Liu, and B. Wang, "Features for link prediction in social networks: a comprehensive study," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, Oct. 2012, pp. 1706–1711. doi: 10.1109/ICSMC.2012.6377983.

[13] L. Dong, Y. Li, H. Yin, H. Le, and M. Rui, "The algorithm of link prediction on social network," *Mathematical Problems in Engineering*, vol. 2013, pp. 1–7, 2013, doi: 10.1155/2013/125123.

[14] C. A. Bliss, M. R. Frank, C. M. Danforth, and P. S. Dodds, "An evolutionary algorithm approach to link prediction in dynamic social networks," *Journal of Computational Science*, vol. 5, no. 5, pp. 750–764, 2014, doi: 10.1016/j.jocs.2014.01.003.

[15] J. Tang, S. Chang, C. Aggarwal, and H. Liu, "Negative link prediction in social media," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, New York, NY, USA: ACM, Feb. 2015, pp. 87–96. doi: 10.1145/2684822.2685295.

[16] K. Zhou, T. P. Michalak, T. Rahwan, M. Waniek, and Y. Vorobeychik, "Attacking similarity-based link prediction in social networks," *arXiv*, Sep. 2018.

[17] C. M. de Moraes, E. Bezerra, and R. Goldschmidt, "Link prediction in social networks: combining topological and contextual data in a community detection based method," in *Proceedings of the 25th Brazillian Symposium on Multimedia and the Web*, New York, NY, USA: ACM, Oct. 2019, pp. 297–304. doi: 10.1145/3323503.3349556.

[18] S. Mallek, I. Boukhris, Z. Elouedi, and E. Lefèvre, "Evidential link prediction in social networks based on structural and social information," *Journal of Computational Science*, vol. 30, pp. 98–107, Jan. 2019, doi: 10.1016/j.jocs.2018.11.009.

[19] S. Haghani and M. R. Keyvanpour, "A systemic analysis of link prediction in social network," *Artificial Intelligence Review*, vol. 52, no. 3, pp. 1961–1995, Oct. 2019, doi: 10.1007/s10462-017-9590-2.

[20] H. Yuliansyah, Z. A. Othman, and A. A. Bakar, "Taxonomy of link prediction for social network analysis: a review," *IEEE Access*, vol. 8, pp. 183470–183487, 2020, doi: 10.1109/ACCESS.2020.3029122.

[21] H. Kou *et al.*, "Trust-based missing link prediction in signed social networks with privacy preservation," *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–10, Nov. 2020, doi: 10.1155/2020/8849536.

[22] L. Cai, J. Li, J. Wang, and S. Ji, "Line graph neural networks for link prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3080635.

[23] J. Chen, X. Wang, and X. Xu, "GC-LSTM: graph convolution embedded LSTM for dynamic network link prediction," *Applied Intelligence*, vol. 52, no. 7, pp. 7513–7528, May 2022, doi: 10.1007/s10489-021-02518-9.

[24] R. Patel, Y. Guo, A. Alhudhaif, F. Alenezi, S. A. Althubiti, and K. Polat, "Graph-based link prediction between human phenotypes and genes," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–8, Mar. 2022, doi: 10.1155/2022/7111647.

[25] P. Wang, B. Xu, Y. Wu, and X. Zhou, "Link prediction in social networks: the state-of-the-art," *arXiv*, vol. 58, no. 1, pp. 1–38, Nov. 2014, doi: 10.48550/arXiv.1411.5118.

[26] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Evolutionary artificial neural networks by multi-dimensional particle swarm optimization," *Neural Networks*, vol. 22, no. 10, pp. 1448–1462, Dec. 2009, doi: 10.1016/j.neunet.2009.05.013.

[27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, 1995, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.

[28] D. K. Altmemi and B. S. Yaseen, "A new method based on swarm intelligence with encrypted data in wireless sensor networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 3, pp. 1525–1533, Jun. 2023, doi: 10.11591/ijeecs.v30.i3.pp1525-1533.

[29] A. N. S. Shaari, M. S. Hadi, A. M. A. Wahab, R. T. P. Eek, and I. Z. M. Darus, "Active vibration control of flexible beam system based on cuckoo search algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 2, pp. 2289–2298, Apr. 2023, doi: 10.11591/ijece.v13i2.pp2289-2298.

[30] J. Y. Khaseeb, A. Keshk, and A. Youssef, "A hybrid swarm intelligence feature selection approach based on time-varying transition parameter," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 1, pp. 781–795, Feb. 2023, doi: 10.11591/ijece.v13i1.pp781-795.

[31] T. Paruchuri, G. R. Kancharla, and S. Dara, "Solving multiple sequence alignment problems by using a swarm intelligent optimiza-

tion based approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 1, pp. 1097–1104, Feb. 2023, doi: 10.11591/ijece.v13i1.pp1097-1104.
[32] S. Mahmood, N. Z. Bawany, and M. R. Tanweer, "A comprehensive survey of whale optimization algorithm: modifications and classification," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 2, pp. 899–910, Feb. 2023, doi: 10.11591/ijeecs.v29.i2.pp899-910.
[33] S. Tarun, M. K. Dubey, R. S. Batth, and S. Kaur, "An optimized cost-based data allocation model for heterogeneous distributed computing systems," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, pp. 6373–6386, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6373-6386.
[34] I. El Hajjami and B. Benhala, "Radio-frequency circular integrated inductors sizing optimization using bio-inspired techniques," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, pp. 6320–6331, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6320-6331.
[35] H. S. Mehdy, N. J. Qasim, H. H. Abbas, I. Al-Barazanchi, and H. M. Gheni, "Efficient time-series forecasting of nuclear reactions using swarm intelligence algorithms," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 5, pp. 5093–5103, Oct. 2022, doi: 10.11591/ijece.v12i5.pp5093-5103.
[36] M. A. A. Mohammad and M. M. T. Jawhar, "Compare between PSO and artificial bee colony optimization algorithm in detecting DoS attacks from network traffic," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 4, pp. 780–787, Aug. 2022, doi: 10.12928/telkomnika.v20i4.23757.
[37] M. M. Jawad, M. T. Younis, and A. T. Sadiq, "Solving flexible job-shop scheduling problem using harmony search-based meerkat clan algorithm," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, no. 2, pp. 423–431, Jun. 2022, doi: 10.11591/ijai.v11.i2.pp423-431.
[38] H. A. Saleh, R. A. Sattar, E. M. H. Saeed, and D. S. Abdul-Zahra, "Hybrid features selection method using random forest and meerkat clan algorithm," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 5, pp. 1046–1054, Oct. 2022, doi: 10.12928/telkomnika.v20i5.23515.
[39] T. C. Truong, J. Plucar, B. Q. Diep, and I. Zelinka, "X-ware: a proof of concept malware utilizing artificial intelligence," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 2, pp. 1937–1944, Apr. 2022, doi: 10.11591/ijece.v12i2.pp1937-1944.
[40] K. G. Abdulhussein, N. M. Yasin, and I. J. Hasan, "Comparison of cascade P-PI controller tuning methods for PMDC motor based on intelligence techniques," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 1, pp. 1–11, Feb. 2022, doi: 10.11591/ijece.v12i1.pp1-11.
[41] O. Hamed, M. Hamlich, and M. Ennaji, "Hunting strategy for multi-robot based on wolf swarm algorithm and artificial potential field," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 1, pp. 159–171, Jan. 2022, doi: 10.11591/ijeecs.v25.i1.pp159-171.
[42] P. Matrenin *et al.*, "Generalized swarm intelligence algorithms with domain-specific heuristics," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 157–165, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp157-165.
[43] P. V. Matrenin, V. Z. Manusov, N. Khasanzoda, and D. V. Antonenkov, "Application of swarm intelligence algorithms to energy management of prosumers with wind power plants," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 6, pp. 6172-6179, Dec. 2020, doi: 10.11591/ijece.v10i6.pp6172-6179.
[44] H. J. Abd, A. S. Abdullah, and M. S. S. Alkafaji, "A new swarm intelligence information technique for improving information balancedness on the skin lesions segmentation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 6, pp. 5703–5708, Dec. 2020, doi: 10.11591/ijece.v10i6.pp5703-5708.
[45] A. Abuelhaija, A. Jebrein, and T. Baldawi, "Swarm robotics: design and implementation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 2, pp. 2173–2181, Apr. 2020, doi: 10.11591/ijece.v10i2.pp2173-2181.
[46] M. E. J. Newman, "Clustering and preferential attachment in growing networks," *Physical Review E*, vol. 64, no. 2, p. 025102, Jul. 2001, doi: 10.1103/PhysRevE.64.025102.
[47] L. Lü, C.-H. Jin, and T. Zhou, "Similarity index based on local paths for link prediction of complex networks," *Physical Review E*, vol. 80, no. 4, p. 046122, Oct. 2009, doi: 10.1103/PhysRevE.80.046122.
[48] T. Zhou, L. Lü, and Y.-C. Zhang, "Predicting missing links via local information," *The European Physical Journal B*, vol. 71, no. 4, pp. 623–630, Oct. 2009, doi: 10.1140/epjb/e2009-00335-8.
[49] L. A. Adamic and E. Adar, "Friends and neighbors on the Web," *Social Networks*, vol. 25, no. 3, pp. 211–230, Jul. 2003, doi: 10.1016/S0378-8733(03)00009-1.
[50] S. P. Harter, "Introduction to modem information retrieval (Gerard Salton and Michael J. McGill)," *Education for Information*, vol. 2, no. 3, pp. 237–238, Jul. 1984, doi: 10.3233/EFI-1984-2307.
[51] C. Wang and W. Song, "A modified particle swarm optimization algorithm based on velocity updating mechanism," *Ain Shams Engineering Journal*, vol. 10, no. 4, pp. 847–866, Dec. 2019, doi: 10.1016/j.asej.2019.02.006.
[52] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, Dec. 1977, doi: 10.1086/jar.33.4.3629752.
[53] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations," *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, Sep. 2003, doi: 10.1007/s00265-003-0651-y.
[54] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002, doi: 10.1073/pnas.122653799.
[55] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006, doi: 10.1073/pnas.0601602103.
[56] I. Ahmad, M. U. Akhtar, S. Noor, and A. Shahnaz, "Missing link prediction using common neighbor and centrality based parameterized algorithm," *Scientific Reports*, vol. 10, no. 1, p. 364, Jan. 2020, doi: 10.1038/s41598-019-57304-y.

BIOGRAPHIES OF AUTHORS

Deepjyoti Choudhury completed his B.Tech in Information Technology in 2011. He completed his M.Tech in Information Technology in 2013 from Assam University, Silchar, India. Currently, he is working as a research scholar in the Department of Computer Science and Engineering, Assam University. His research area of interest is social network analysis, machine learning, and deep learning. He has published several publications in journals and conferences. He can be contacted at email: deepjyotichoudhury05@gmail.com.

Dr. Tapodhir Acharjee completed his M.Tech in Information Technology from Tezpur University and Ph.D. from Assam University, Silchar, India. He is currently working as Assistant Professor in the Department of Computer Science and Engineering, Assam University. He has more than 15 years of teaching experience. He has authored many peer review journals and conferences. His research area includes network security, social network analysis, and machine learning. He can be contacted at email: tapacharjee@gmail.com.