International Journal of Cloud Computing and Database Management

E-ISSN: 2707-5915
P-ISSN: 2707-5907
IJCCDM 2020; 1(1): 28-36
Received: 13-11-2019
Accepted: 17-12-2019

Amin Rezaeipanah
Department of Computer, University of Rahjuyan Danesh Borazjan, Bushehr, Iran

Mohammad Javad Mokhtari
Technical and Vocational University, College of Bushehr, Bushehr, Iran

Milad Boshkani Zadeh
Department of Computer Engineering, Ahram Branch, Islamic Azad University, Ahram, Iran

Corresponding Author:
Amin Rezaeipanah
Department of Computer, University of Rahjuyan Danesh Borazjan, Bushehr, Iran

Providing a new method for link prediction in social networks based on the meta-heuristic algorithm

Amin Rezaeipanah, Mohammad Javad Mokhtari and Milad Boshkani Zadeh

DOI: https://doi.org/10.33545/27075907.2020.v1.i1a.6

Abstract
Social network analysis is one of the most important research fields in data mining today. The purpose of the analysis of these networks is to extract the embedded knowledge in the data set and to learn the behavior of users in the social networking environment. One of the most attractive and central applications of social network analysis is link prediction. The purpose of link prediction in social networks is to identify missing and unknown information from users or to predict the future link between two users. In recent years, various artificial intelligence algorithms have been introduced as one of the most important tools for resolving link prediction and big data. In this research, a strategy based on a meta-heuristic is used to improve link prediction in social networks. The proposed method is based on the characteristics of the signed social networks provided and turns the link prediction problem into a two-class classification problem. Then uses the capability of the Particle Swarm Optimization (PSO) and the topological properties of the social network graph to create a database with two classes, the first class pointing to the existence of a connection between the users and the second class pointing to the absence of this relationship. Creates a database using the support vector machine model for categorization work and uses the classic Katz similarity criterion for end-user suggestions. Twitter social network information has been used to compare and evaluate the proposed method. The results of the experiments show the superiority of the proposed method with 0.23, 0.99, and 6.32, respectively, compared to Meta-Path, Katz and CN algorithms in F-measure criterion.

Keywords: Social networks, link prediction, meta-heuristic algorithms, similarity criteria

1. Introduction

Social networks are defined as social structures of individuals that are directly or indirectly linked to each other through a series of relationships such as friendship, file sharing, emailing, and so on. Facebook and Twitter are important examples of these networks. Social networking is a popular way to model interactions between people in a group or community. These networks are the result of a combination of different sciences concepts, especially social sciences and computer science [1]. With the development of virtual social communities, network analysis and community extraction have also evolved. Social networks can be visualized as graphs, where, a node belongs to a person in a group and an edge represents a kind of relationship between related people [2]. Social networks are very dynamic as new edges and spheres are added to the graph over time [3, 5]. Understanding the dynamics that evolve the social network is a complex problem due to the large number of variable parameters. But a relatively simpler problem is to understand the relationship between two particular nodes. For example, some of the interesting questions that can be asked are: How do communication patterns change over time? What are the factors that govern the community? How is the connection between two nodes affected by the other nodes?

In fact, the main issue under consideration in this study is to predict the probability of future connections between two nodes, assuming that there is no relationship between the nodes in the current state of the graph. This problem is commonly known as link [6] prediction. If we show the social network with a graph $G = (V,E)$ where V is a set of nodes and E is a set of graph edges, then the problem of link prediction is to identify the set E' They are not present in the G graph but may be added to the G graph at some time in the future [7]. These edges form a complex graph with the maximum number of edges possible in Eq. (1).

$$U = \frac{|V| \times (|V| - 1)}{2} \quad (1)$$

In such an arrangement, the set of available edges E and the set of edges that may be added in the future (E') do not share $(E \cap E' = \emptyset)$. Link prediction specifies the edges of the set E' using the properties of the graph framework in social networks. Also $E + E' = U$ and link prediction try to find the set E' for example $E' = U - E$.
In continuation of this research, the classical similarity criteria are presented in Section 2. In the Section 3, we review some of the most related work on link prediction in social networks. The proposed method is discussed in Section 4 and the results are presented in Section 5. Finally, Section 6 conclusion.

2 Similarity Criteria

In this section, we review some of the most famous criteria used empirically in link prediction. Similarity criteria are used to calculate the (relationship) of future links between two users. These criteria are based on graph topology.

2.1 Katz Criterion

The Katz index uses the sum of all paths between node pairs in graph G exponentially by the length of short paths to calculate similarity [8]. This criterion calculates the similarity between the nodes u and v with Eq. (2).

$$Score(u, v) = \sum_{l=1}^{\infty} \beta^l . |paths_{u,v}^l| \quad (2)$$

Where, the constant coefficient β is used to reduce the effect of long paths in determining a link score. The use of β^l in this respect will increase the calculated value for link prediction in shorter paths. $|paths_{u,v}^l|$ Also, the number of paths l is between two users u and v.

2.2 Friend of friend criterion

The criterion of a friend of friend depends on the number of friends that share the nodes u and v, as shown in Eq. (3) [9].

$$Score(u, v) = \Gamma(u) \cap \Gamma(v) \quad (3)$$

In this respect $\Gamma(u)$ and $\Gamma(v)$ are the sum of the neighbors u and v.

2.3 Common Neigh bor criterion

The shared neighbors index measures the number of shared neighbors between u and v. This criterion has proven to be a good indicator of the possibility of future links occurring in a collaboration network with a number of common neighbors Eq. (4) shows how to calculate this criterion [10].

$$Score(u, v) = |\Gamma(u) \cap \Gamma(v)| \quad (4)$$

2.4 Friend link criterion

The friendlink criterion measures the similarity criterion according to the frequency and path length between two nodes u and v Eq. (5) represents the criterion of friendlink [11].

$$Score(u, v) = \sum_{i=2}^{l} \frac{1}{i-1} . \frac{|paths_{u,v}^i|}{\prod_{j=2}^{i}(n-j)} \quad (5)$$

Where, n is the number of nodes l is the maximum length of a given path between the nodes u and v. Term $1/(i - 1)$ is a attenuating factor that determines the weight of paths with respect to length l. $|path_{vi,vj}^i|$ The number of all paths of length l is between u and v. $\sum_{j=2}^{i}(n-j)$The total number of possible paths between u and v if each node of the graph has links to all other nodes.

2.5 Adamic-Adar Criterion

Adamic-Adar index proposed by Adamic and Adar is calculated by adding weights to the nodes which are connected to both nodes u and v [11]. Eq. (6) shows how to calculate this criterion.

$$Score(u, v) = \sum_{z \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log k_z} \quad (6)$$

Where, z is a common neighbor to nodes both u and v and k is the degree of node z.

2.6 Jaccard Criterion

The jaccard index measures the probability that a neighbor of u and v is a neighbor of both u and v nodes. This measurement is a method of defining shared content properties and is important in retrieving information Eq. (7) Illustrates this criterion [12].

$$Score(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|} \quad (7)$$

3 Related Works

Police *et al*. (2014), have proposed a Covariance Matrix Adaptation Strategy (CMA-ES) for predicting links on Twitter's social network [10]. The scoring matrix is constructed using 16 different similarity criteria and it uses the evolutionary algorithm method to find the optimal coefficients.
Al Hassan *et al*. (2006), demonstrated that using external data outside the range of graph topology can significantly improve link prediction results in social networks [13]. They also used different similarity criteria as a feature in a supervised learning program in which the link prediction problem is posed as a binary classification problem. Since then, the supervised classification method has gained popularity in link predictions [14]. Dopa *et al*. (2010), proposed a learning algorithm for predicting transplantation based on possible limitations [15]. They performed the

learning with the feature vectors calculated from the training data and then used this model to predict future links. Lee *et al* (2018), have proposed a method for link prediction in heterogeneous networks based on back propagation neural network [16]. They have used cross-path features in their method. In this research, after extracting these features, a supervised link prediction model is developed using a three-layer back propagation neural network.

Moradabadi and Meybodi (2018), presented a method based on learning Atomata to predict the link between two nodes in the future [17]. Their study focused on Atomata's learning strategy on weighted social networks. Their purpose is to estimate the weight of each test link using other weights in the network. Torabi *et al* (2016), have proposed a hybrid classification method for link prediction on location-based social networks [18]. They divided the network nodes into eight groups according to their position and then concluded that the mean nodes in each group above 600 nodes were those of high density.

Srilata *et al*. (2017), They have presented a method based on the firefly algorithm for link prediction on social networks [19]. They used a search method for fireflies to predict the link. The similarity score between the node pairs is calculated at the end of the iterations, and the node pair with the highest score is likely to form a link in the future. In another study, Sun *et al*. (2015), proposed a systematic social approach containing social network information to create a referral system [20]. In this research, user-friendliness records are used to predict missing links in the user matrix.

Stephen *et al*. (2017), presented a comparative study between methods of calculating similarity in a memory-based filtering recommender system [21]. This study aims to increase the understanding of user interest in the broad domain in order to provide better recommendations as well as to reduce Spartan data. In a similar study by Sperley *et al*. (2018), they presented research on suggesting a system to improve the approach of social networks [22]. In this research, a recommendation system for big data applications designed to provide useful recommendations on online social networks is used. The proposed technique is a collaborative and user-centric approach that exploits the interactions between users and creates multimedia content on one or more social networks in a new and effective way.

Madhali *et al*. (2019), used graphical feature analysis (user profiles) to predict link in social networks [23]. They have stated that there are generally three ways to solve the bond prediction problem: feature-based models, bayesian probabilistic models, and probabilistic relational models. In feature-based methods, graphic features are extracted and used for classification. These features are usually divided into three groups: neighborhood-based, path-based, and node-based. This research confirms the extraction of feature-based topological information that has not been addressed in previous studies. Finally, the results of the comparison of the extracted features show that the clustering coefficient and the shortest path are effective in predicting the link with the accuracy criterion of 0.97.

Aghabozorgi and Khayyambashi (2018), proposed a new similarity criterion for predicting graft based on local structures in social networks [24]. This similarity criterion is presented through a two-step peer learning model. Firstly, a large database containing social network edges is created by measuring the similarity of the vertex and destination vertices. This database also contains structural information related to vertices. Secondly, a predictive learning model based on transplant prediction is presented. The accuracy of this Highschool method is reported to be 93%.

Flage *et al*. (2016), used graph theory and user profile to socialize on social networks, In the first stage of the proposed system, all users are clustered according to the structural similarity and profile using the K_Medoids method [25]. Secondly, the friendLink algorithm is applied to the users of each cluster and the similarity between the users is calculated. The results of the experiments show the F-criterion in the link prediction for Google+ social network is about 0.3 and for Last.fm social network is about 0.25. Chen and Chen (2014), used the ant colony optimization algorithm and shared neighbor similarity index to predict missing links in the social network [26]. In [27] presented an algorithm for predicting ant colony link optimization (ACOLP). In this method, the triangular structures are predicted using ant colony optimization and then the substructures of the graphs that may generate links in the future are predicted.

Jalili *et al*. (2017), developed a Meta-Path algorithm for link prediction in multiple social networks based on categorization models [28]. This algorithm is meta-path based and uses three methods NB, SVM and KNN to categorize. In the meta-path definition, u and v are two users on multiple social networks. A meta-path of length 2 from u in the network like Twitter to v in the network like quadrilateral $u \xrightarrow{follows} w \xrightarrow{friends} v$ It has been shown that the relationship between u and v is considered; u is looking for w in Twitter and w is in love with v in quadrilateral. In order to define the features associated with the link prediction problem in the categorization algorithm, here the length and type of communication in the meta-path between two users is determined.

4 Proposed Method

This research provides an optimal way to predict links in social networks with a meta-heuristic approach. The proposed method is derived from extracting topological graph widgets based on ego-based social networks. Ego-centered social networks are a type of social network that measures and portrays inter-user communication through networks called ego. In visualizing the graph of such networks, the ego user is usually at the center of the graph. The proposed method is based on the characteristics of the tagged social networks and converts the link prediction problem into a two-class classification problem. It then uses the capability of the particle swarm optimization and the topological features of the social network graph to create a database with two classes, the first class pointing to the existence of a connection between the users and the second class pointing to the absence of this relation. Support Vector Machine (SVM) algorithm is used here to classify. Creating a database is based on each pair of users. For this purpose, each user pair will have a sample in the database. The class of this sample is based on social network information. If there is a link between these two users, the "connection" label and otherwise the "non-connection" label will be assigned.

Real social networks are usually very sparse and this reduces the efficiency of categorization algorithms. Because

a social network has a large number of users and a scale user relates to the total number of users with only a small number of them. This will cause the data in the database to be highly disordered. In general, this disorder is due to an increase in instances of the non-connection class compared to the connection class in the database. This is a major challenge for the proposed algorithm, which is attempting to reduce its negative impact by selecting part of the "non-connection" class samples by the particle swarm optimization. The proposed meta-heuristic method is improved by integrating similar and closely related samples from the non-connection class before applying the particle swarm optimization.
Creating a database uses the support vector machine model for categorization work and the classic Katz similarity criterion for end-user suggestions.

4.1. Pre-processing and Signed Social Network

In this section users are training sections (E^T) and experiment (E^P) are determined. Here, user segmentation is based on 10-Fold validation method. After that, the orientation of the social network structure is modified by signal allocation. Between both users who have followed at least one other, a positive sign for the follower and a negative sign for the opposite are assigned. Figure 1 shows how to sign the social network.

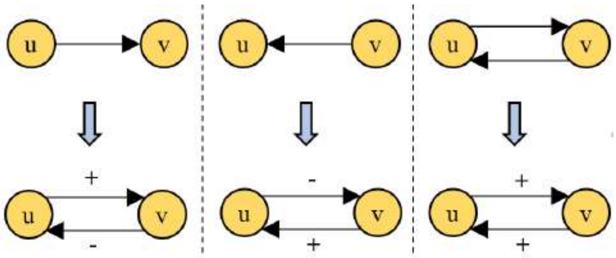

Fig 1: Allocation of sign to social networking links

4.2 Feature Extraction

To build a database, several topological features are extracted from the social network graph. Here are six features extracted for both u and v in training users.

1. *Reputation feature for user u:* This feature is presented in [28] and is related to the popularity of user u in a signified social network. $RP(u)$ the normalized reputation feature is u user and is defined as Eq. (7).

$$RP(u) = \frac{d_{in}^{+}(u) - d_{in}^{-}(u)}{d_{in}^{+}(u) + d_{in}^{-}(u)} \tag{7}$$

Where, $d_{in}^{+}(u)$ and $d_{in}^{-}(u)$ the number of input links is "positive" and "negative" for user u, respectively.

2. *Reputation feature for user v:* This feature is calculated similar to the reputation feature for user u. $RP\ (v)$ is the normalized reputation v user feature.

3. *Optimism feature for user u*: Optimism another structural feature of graph topology is presented in [28]. $OP(u)$ is the normalized optimistic feature of the user u and is defined as Eq. (8).

$$OP(u) = \frac{d_{out}^{+}(u) - d_{out}^{-}(u)}{d_{out}^{+}(u) + d_{out}^{-}(u)} \tag{8}$$

Whrer, $d_{out}^{+}(u)$ and $d_{out}^{-}(u)$ The number of output links is denoted by "positive" and "negative" respectively.

4. *Optimism feature for user v*: The value of $OP(v)$ is also defined as the Optimism feature for user u.

5. *Shared tweets feature:* This feature is calculated based on the number of similar keywords used in the two-user tweets Eq. (9) defines the value of this feature for two users u and v.

$$CT(u, v) = |feat_u \cap feat_v| \tag{9}$$

Where, $feat_u$ and $feat_v$ the keywords used are the tweets of the users u and v, respectively.

6. *Shared circles feature:* In a ego-centered social network, users interact with each other through paths that cross different network circles. The value of the shared circles feature for the two users u and v is calculated by the number of Ego-Paths with the shared circles between the two users. Ego-Path $u \xrightarrow{follows} w \xrightarrow{friends} v$ with length 2 defined, where user w communicates between two users u and v. A Ego-Path can be provided in varying lengths, so that the user similar to w must be at least one of the u or v users in the same circle of an ego-network. The feature of shared circles between users u and v is represented as $CC\ (u, v)$, and is defined as the number of similar users w for each path along the path shared by at least one user u or v. The lengths considered are 2, 3 and 4. Figure 2 shows an example of this feature.
This example contains three circles in an ego- network and two Ego-Paths of length 2. The first path is $u \rightarrow w \rightarrow v$ where, user w is in the shared circle (circle 1) with user v. The second path is also $x \rightarrow y \rightarrow z$ where, user y is associated with user x in the shared circle (circle 2).

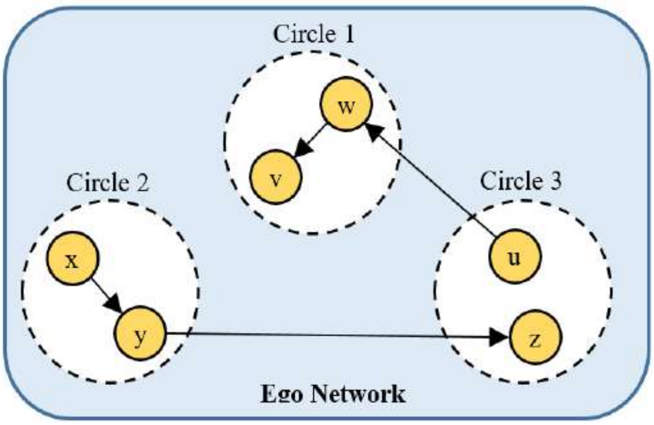

Fig 2: An example of a shared circles feature

4.3. Creating Database

The extracted topological features are used to create a database with two classes, where the first class refers to the existence of a connection between the users and the second class refers to the absence of such a connection. Database instances are created based on each user pair. For this purpose, each user pair will have a sample in the database. The class of this sample is based on social network information. If there is a link between these two users, the connection label and otherwise the non-connection label will be assigned.

Given N users, about N^2 samples will be created per user. This large number of samples leads to the creation of a large database due to the large number of users. Furthermore, the number of instances with the non-communication label is much higher than the communication label. To address these two challenges, the process of integrating similar samples as well as reducing data scattering has been used.

4.4 Merging Similar Samples

In the database created many examples are seen with great similarity. Hence, database instances that are less than the threshold value θ Merge. $D(I,j)$ denotes the distance between two samples i and j and is defined by Eq. (10).

$$D(i,j) = \frac{1}{M}\sum_{k=1}^{M} |f_k(i) - f_k(j)| \tag{10}$$

Where, $f_k(i)$ and $f_k(j)$ are the k-th feature values in i and j samples, respectively. Also, M is the number of samples extracted. Given this distance, the two samples can be merged only if their class label is shared. The merging of the two samples is performed as the mean value for each feature.

4.5 Reduce Database Dispersion

Real social networks are usually very sparse, and this reduces the efficiency of categorization algorithms. Because there are so many users on a social network, and a single-scale user has only a small number of links to all social network users. To solve this problem, particle swarm optimization in the form of a meta-heuristic algorithm is used to select and remove instances of the non-communication label to reduce data scatter. The steps of this algorithm are as follows:

1. *Particles structure:* The structure of the position of each particle in the search space according to the purpose of the problem is shown in Figure 3.

s_1	s_2	$\cdots$	s_i	$\cdots$	s_w

Figure 3: Particle position structure

In this structure, s_i represents the i-th instance of the database in the non-communication class and is defined as binary. Also, w the total number of instances is associated with a label of non-communication. Given this structure, each solution is defined as a binary string in which 0 means the sample is removed from the database and 1 represents the sample used.

2. *Initial population:* Due to the particle structure of the initial population, a random population is created.

3. *Objective Function:* The objective function of the problem is defined by entropy. The entropy in a database indicates the degree of irregularity (scatter). The index for a database with two classes is at least zero when half of the samples are class 1 and the other half are class 2. Furthermore, when all samples belong to only one class, the entropy becomes maximum (value 1). Therefore, low entropy represents less dispersion in the database, thus providing more suitable conditions for classification models. Hence, the entropy-based objective function as well as the accuracy of the SVM classification model are computed as Eq. (11).

$$F(P_i) = \frac{ACC(S,P_i)}{I(S,P_i)} \tag{11}$$

Where, $F(P_i)$ i-th particle fit, $I(S,P_i)$ and $ACC(S,P_i)$ the entropy and accuracy of SVM classification are respectively in the S database with respect to the samples selected in P_i.

4. *Parameters:* The velocity vector (V) for each particle is initially randomly assigned to the position vector dimension. In addition, the $gBest$ parameter indicates the best position of the particle to achieve it, and $gBest$ indicates the best position ever found by the particle population.

5. *Update particle velocity and position*: In each iteration of the algorithm, the velocity and position of each particle are updated based on the parameters $pBest$ and $gBest$ according to Eq. (12) and Eq. (13).

$$V_{k+1} = V_k + c_1 \times r_1 \times (pBest_k - P_k) + c_2 \times r_2 \times (gBest - P_k) \tag{12}$$

$$P_{k+1} = \langle P_k + V_k \rangle \tag{13}$$

Where, V_k and P_k are the current velocity and position of the k-th particle, c_1 and c_2, respectively, of the learning constant parameters, and r_1 and r_2 are random numbers between $[0,1]$. $pBest_k$ is the best k-th particle position and $gBest$ is the best global position in the population. In this paper, due to the discrete environment, the term $\langle P_k + V_k \rangle$ is used to update the position of the particles. This term is considered a round number for each sample. The minimum and maximum speed and position ranges are reviewed after the update. The minimum and maximum speeds are set in the range [0, 1] and the position vector always has one of two values of 0 or 1.

6. *Termination condition*: In this research, the algorithm's stopping condition is the number of itrations constant.

4.6 Link Prediction Process

The link prediction process for each user is tested individually according to the classification model. For each test user, Top-K user is suggested from the training section. The link prediction process is done in 5 steps:

1. The target user features are extracted with all training users.
2. A small database is created for the target user according to the extracted features.

3. Similar examples are merged into the non-communication label.
4. The SVM classification model is applied to the database and the training users are selected with the predicted communication label.
5. Ranking of selected users is done by Katz similarity criterion and Top-K user with highest ranking is recommended to target user.

5. Results and Discussion

In this paper, extensive experiments are presented to demonstrate the superiority of the proposed algorithm. The results are compared against the classical similarity criteria of Katz, Common Neighbor (CN) and Meta-Path algorithm. MATLAB version 2017a software was used to perform the simulation, which allows its toolboxes to analyze the results of the proposed method more closely. The simulation was performed using a PC with a configuration of Intel 5 Core processor and 2.4GHz frequency, 16GB memory and Windows 10 operating system.

To be sure, all comparisons use the same social network and parameters, and in order to present more accurately, the results are reported as averaging 10 distinct performances. In addition, the 10-Fold method has been used for validation. The proposed particle swarm optimization is configured with a population of 35 and a maximum of 200 iterations. Other parameters used in the simulation are as follows:

$$\theta = 0.15, \beta = 0.05, c_1 = 0.6, c_2 = 0.4$$

Twitter's ego-centric social network dataset is used to simulate the information. This dataset is freely accessible from the SNAP website. The Twitter dataset consists of a number of ego-networks, each ego-network having one ego-user and a number of circle. The circles show a list of users that are distributed around the ego-user. Given the sheer volume of Twitter social networking data and the high number of ego-networks, users, and connections between them, this study used only part of this information for comparison and evaluation purposes. Here are 20 elementary ego-networks with 2101 nodes and 34023 links.

Information available from any ego-network includes "Directional Links among Users", "Interlinks to Users" and "Users Tweets". We use different criteria and standards of Precision, Recall and F-Measure to evaluate and compare results. These criteria are calculated for each user based on the list of actual related users and the list of predicted relevant users. The precision criterion calculates the ratio of the number of related users from the Top-K list to the number of Top-K users. The recall criterion calculates the ratio of the number of related users from the Top-K list to the total number of related users, and the f-measure is the normalized harmonic mean of the precision and recall criteria.

Comparisons are made according to two lists of true friend users and suggested friend users. In addition, tests for each criterion are conducted in two modes: user-based and number-based. User-based means that each evaluation criterion is calculated and presented based on the size of the Top-K list for each test user. Also, user-based on the number of suggestions, in the sense of calculating each evaluation criterion for all training users with a specified Top-K size, and reporting each criterion as an average. In the first experiment, the Top-K is set to 10 and the precision criterion for each test user is reported in Figure 4.

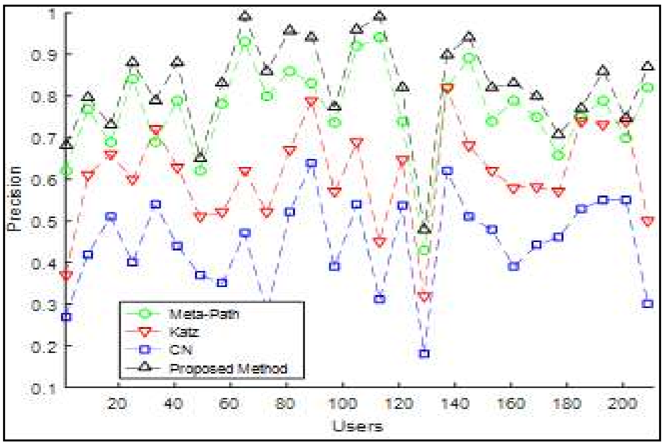

Fig 4: Precision criterion results for each user

The results of this comparison show the superiority of the proposed method for most users. Following the proposed method, the Meta-Path, Katz and CN algorithms are in the next rank. In another experiment, the Top-K values of 1 to 20 were averaged for all test users. This provides a better performance than the precision criterion. Figure 5 shows the results of this comparison.

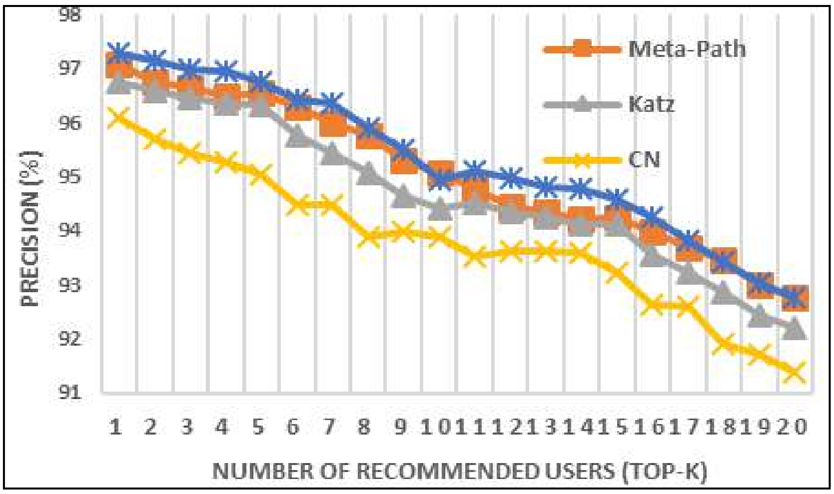

Fig 5: Precision criterion results with different proposed users

The results show the tangible superiority of the proposed method in most Top-K. Except for numbers 10, 18, 19 and 20, in other cases the proposed method performs better than the Meta-Path algorithm. At best, the Precision criterion for the proposed method is 97.3% and this is achieved with Top-K = 1. In the next experiment, the results of the comparison of the recall criterion in figure 6 are presented for each test user and Top-K = 10.

Similar to the precision criterion, this recall criterion also affects the accuracy of the Top-K. Therefore, the results of this criterion for different methods and Top-K of 1 to 20 are investigated. Figure 7 shows the results of this comparison.

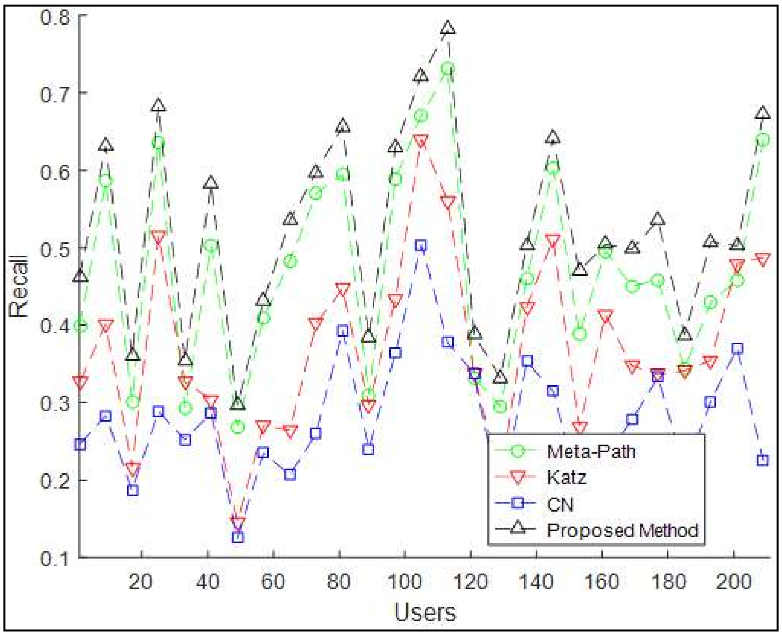

Fig 6: Recall criterion results for each user

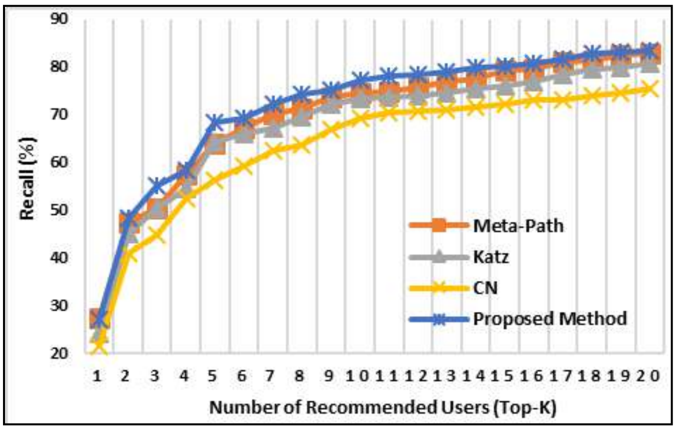

Fig 7: Recall criterion results with different proposed users

The results of this comparison show that in all different numbers of Top-K except 2 and 20, the proposed method is superior to all methods. Following the proposed method, the Meta-Path, Katz and CN algorithms are in the next rank, respectively. In the average recall criterion in the proposed method, the number of proposed users is 20 to 83.31%. However, at best, sometimes the accuracy of the proposed method is as high as 88%. Finally, Figure 8 shows the results of comparing the f-measure criterion for the proposed method and the Meta-Path, Katz and CN methods. This test is tested for each user based on Top-K = 10.

In another experiment to examine more closely, the results of the f-measure criterion for the number of different users proposed are examined. Here are the suggested number of users from 1 to 20 in Figure 9.

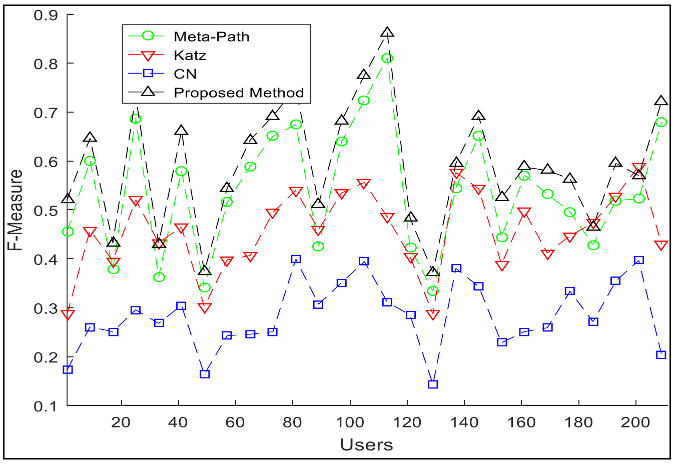

Fig 8: F-measure criterion results for each user

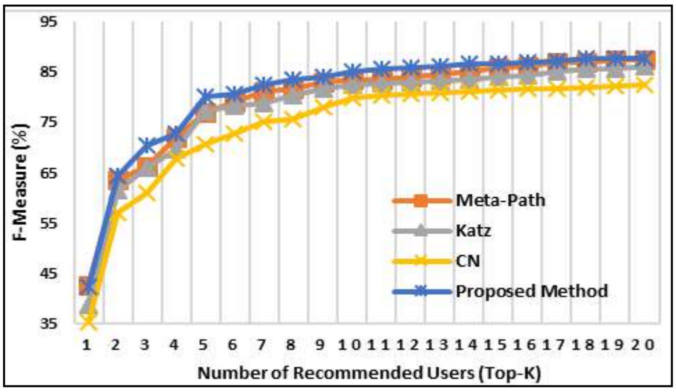

Fig 9: F-measure criterion results with different proposed users

The results of this comparison clearly show the superiority of the proposed method. The best performance of the proposed method is obtained with Top-K = 18 and 87.77% accuracy. After the proposed Meta-Path method with Top-K = 20 and accuracy of 87.59% is in second place. Katz and CN algorithms are next in the rankings with 86.07% and 82.57%, respectively. In general, the results show the superiority of the proposed method in most experiments and for some criteria. In other cases, the proposed method has reported acceptable results.

Comparison results show that, in most cases, both the CN and Katz methods perform less well than the proposed methods and Meta-Path. The reason for this is the characteristic of this algorithm. In CN, only the number of neighbors and in Katz the frequency and length of the route are taken into account and no attention is given to the characteristics of ego-centered social networks. It can be concluded that these two algorithms that work only on the basis of local graph properties are not very suitable for ego-centered social networks.

6 Conclusion

In this paper, a combined method based on meta-heuristic algorithms is proposed to improve the link prediction problem on Twitter social network. The particle swarm optimization was used to create a two-class database of topological features of the social network graph and the support vector machine classification algorithm to model this database. In general, the use of different local and topological features extracted is one of the advantages of the proposed method. Other advantages of this advantage include the structure of the particle swarm optimization in reducing database data scattering. Taking into account the shared tweets of users has also been effective in increasing the accuracy of the proposed method. In general, the superiority of the proposed method in f-measure criterion against Meta-Path, Katz and CN methods are 0.23, 1.99 and 6.32%, respectively.

Reference

1. Acquisti A, Gross R. Predicting social security numbers from public data. Proceedings of the National academy of sciences. 2009; 106(27):10975-10980.
2. Barabâsi AL, Jeong H, Néda Z, Ravasz E, Schubert A, Vicsek T, *et al*. Evolution of the social network of scientific collaborations. Physica A: Statistical mechanics and its applications. 2002; 311(3):590-614.
3. Papadimitriou A, Symeonidis P, Manolopoulos Y. Fast and accurate link prediction in social networking systems. Journal of Systems and Software. 2012; 8(9):2119-2132.
4. Papadimitriou A, Symeonidis P, Manolopoulos Y. Fast and accurate link prediction in social networking systems. Journal of Systems and Software. 2012; 85(9):2119-2132.
5. Kossinets G, Watts DJ. Empirical analysis of evolving social networks. Science. 2006; 311:88-90.
6. Szczepański PL, Barcz AS, Michalak TP, Rahwan T. The game-theoretic interaction index on social networks with applications to link prediction and community detection. In Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.
7. Liu F, Liu B, Sun C, Liu M, Wang X. Deep belief network-based approaches for link prediction in signed social networks. Entropy. 2015; 17(4):2140-2169.
8. Katz L. A new status index derived from sociometric analysis. Psychometrika. 1953; 18(1):39-43.
9. Al Hasan M, Zaki MJ. A survey of link prediction in social networks. In Social network data analytics, Springer US, 2011, 243-275.
10. Bliss CA, Frank MR, Danforth CM, Dodds PS. An evolutionary algorithm approach to link prediction in dynamicsocial networks. Journal of Computational Science. 2014; 5(5):750-764.
11. Papadimitriou A, Panagiotis S, Yannis M. Fast and accurate link prediction in social networking systems. Journal of Systems and Software. 2014; 85(9):2119-2132.
12. Niwattanakul S, Singthongchai J, Naenudorn E, Wanapu S. Using of Jaccard coefficient for keywords similarity. In Proceedings of the International
13. MultiConference of Engineers and Computer Scientists. 2013; 1(6).
14. Al Hasan M, Chaoji V, Salem S, Zaki M. Link prediction using supervised learning. In SDM06: workshop on link analysis, counter-terrorism and security, 2006.
15. Bilgic M, Namata GM, Getoor L. Combining collective classification and link prediction. In Seventh IEEE International Conference on Data Mining Workshops ICDMW, 2007, 381-386. IEEE.
16. Doppa JR, Yu J, Tadepalli P, Getoor L. Learning algorithms for link prediction based on chance constraints. In Joint european conference on machine learning and knowledge discovery in databases, 2010, 344-360. Springer, Berlin, Heidelberg.
17. Li JC, Zhao DL, Ge BF, Yang KW, Chen YW. A link prediction method for heterogeneous networks based on BP neural network. Physica A: Statistical Mechanics and its Applications. 2018; 495:1-17.
18. Moradabadi B, Meybodi MR. Link prediction in weighted social networks using learning automata. Engineering Applications of Artificial Intelligence. 2018; 70:16-24.
19. Torabi N, Shakibian H, Charkari NM. An ensemble classifier for link prediction in location based social network. In 2016 24th Iranian Conference on Electrical Engineering (ICEE), 2016, 529-532. IEEE.
20. Srilatha P, Manjula R. Structural similarity-based link prediction in social networks using firefly algorithm. In 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), 2017, 560-564. IEEE.
21. Sun Z, Han L, Huang W, Wang X, Zeng X, Wang M, *et al*. Recommender systems based on social networks. Journal of Systems and Software. 2015; 99:109-119.
22. Stephen SC, Xie H, Rai S. Measures of Similarity in Memory-Based Collaborative Filtering Recommender System: A Comparison. In Proceedings of the 4th Multidisciplinary International Social Networks Conference on ZZZ, 2017, 32. ACM.
23. Sperlì G, Amato F, Mercorio F, Mezzanzanica M, Moscato V, Picariello A, *et al*. A Social Media Recommender System. International Journal of Multimedia Data Engineering and Management (IJMDEM), 2018; 9(1):36-50.

24. Madahali L, Najjar L, Hall M. Exploratory Factor Analysis of Graphical Features for Link Prediction in Social Networks. In International Workshop on Complex Networks, 2019, 17-31. Springer, Cham.
25. Aghabozorgi F, Khayyambashi MR. A new similarity measure for link prediction based on local structures in social networks. Physica A: Statistical Mechanics and its Applications. 2018; 501:12-23.
26. Parvazeh F, Harounabadi A, Naizari MA. A Recommender System for Making Friendship in Social Networks Using Graph Theory and users profile. Journal of Current Research in Science. 2016; (1):535.
27. Chen B, Chen L. A link prediction algorithm based on ant colony optimization. Applied Intelligence. 2014; 41(3):694-708.
28. Sherkat E, Rahgozar M, Asadpour M. Structural link prediction based on ant colony approach in social networks. Physica A: Statistical Mechanics and its Applications 419, 2015, 80-94.
29. Jalili M, Orouskhani Y, Asgari M, Alipourfard N, Perc M. Link prediction in multiplex online social networks. Royal Society Open Science. 2017; 4(2):160863.