Link Prediction in Fuzzy Social Networks using Distributed Learning Automata

Behnaz Moradabadi[1] and Mohammad Reza Meybodi[2]
[1,2] Department of Computer Engineering, Amirkabir University of Technology, Tehran, Iran
[1]moradabadi@aut.ac.ir
[2]mmeybodi@aut.ac.ir

Link prediction is a social network research area that tries to predict future links using the social network structure. This paper proposes a novel link prediction method (FLP-DLA) that is based on fuzzy social networks and Distributed Learning Automata (DLA). Distributed learning automata are reinforcement-based optimization tools which try to learn and converge to the optimal behavior from environmental feedbacks using graph navigation. In the preprocessing phase of the FLP-DLA, the proposed method tries to calculate a fuzzy strength for each link based on the information of the network such as event time. In the main phase of the FLP-DLA, it uses these fuzzy strengths in addition to DLA to find the strength of test links. In each iteration of the proposed method, the DLA tries to find a path between the endpoints of a random test link and then the FLP-DLA calculates the fuzzy strength of the obtained path using the fuzzy strengths of the links through the path and rewards or penalizes the DLA based on the path strength. The main phase is repeated until the LAs converge to some action. Finally, we use the strength of the test links as the output of link prediction. The results reported in this paper have proven satisfactory, indicating the usefulness of the proposed method for some social network data sets.

Keywords: Social Network, Fuzzy Graph, Learning Automata.

I. INTRODUCTION

Nowadays social network helps people and organization to exchange information more efficiently. A social network can be considered as a graph where the nodes are individuals and the edges display some communications between the corresponding individuals, such as collaboration, friendship, like, dislike, … [1] [2]. Since this kind of network is generally complex and highly dynamic, it is really important to understand its behavior along time [1] [3]. To understand the dynamics of social networks, there is a research area called Social Network Analysis (SNA) and link prediction is one of the main tasks undertaken by SNA [1] [4] that focuses to predict what communications can be generated in the future. Link prediction is applicable to a wide variety of application areas. For example, in the area of the Internet and web science, it can be used in tasks like automatic web hyper-link creation and web site hyper-link prediction [1]. In e-commerce, one of the most prominent usages of link prediction is to build recommendation systems [5]. It also has various applications in other scientific disciplines. For instance, in bibliography and library science, it can be used for deduplication and record linkage [6]; in Bioinformatics, it has been used in protein-protein interaction (PPI) prediction [7]. In security related applications, it can be used to identify hidden groups of terrorists and criminals [2].

All link prediction methods address the following question: “Given a pair of nodes u and v in the current social network, how likely is it that u will interact with v in the future?”. To solve a link prediction problem, there are several techniques that the literature introduces them [8]: the most common approach is based on applying topological similarity metrics to non-existence links at time t to determine if a link will appear at a time t’ (t’ > t). Such methods generate scores for each link and use the scores to perform prediction task either by an unsupervised or a supervised technique. There are many similarity metrics that will be briefly explained in section II. The major challenge of standard link prediction methods is that they don’t consider the network evolution or repeated links because they only use the current network structure without considering the links’ frequency and creation time. Our work tries to overcome these problems by using both the fuzzy concept to model the strength of links and DLA to predict future links. The novelty of the proposed method is that it uses the fuzzy concept to present the link strength.

A learning automaton is an adaptive decision-making unit that improves its performance by learning how to choose the optimal action from a finite set of allowed actions through repeated interactions with a random environment [9]. Distributed learning automata (DLA) are networks of interconnected learning automata which collectively cooperate to solve a particular problem. In each iteration, an automaton randomly chooses one of its outgoing edges (actions) according to its action probabilities and activates the learning automaton at the other end of the selected edge. The activated automaton in turn randomly selects an action which results in the activation of another automaton. The process of choosing the actions and activating automata is continued until a criterion is reached. The chosen actions, along with the path, are applied to the random environment. The environment evaluates the applied actions and generates a reinforcement signal to the LAs. The LAs along the chosen path update their action probability vectors on the basis of the reinforcement signal by using a learning algorithm. These steps are repeated until the LAs converge to some action.

In this paper, to predict the future links a new link prediction which uses both fuzzy social network and DLA (FLP-DLA) is proposed. The proposed method has two steps: in the initialization phase, it models the strength of each link as a fuzzy variable by using the time and frequency information of the link. So, we have a fuzzy social network where each link has a fuzzy strength and use this network as the input of next step. The main phase of FLP-DLA uses distributed learning automata to predict the future links based on graph navigation. To do this, in FLP-DLA there is one LA for each node in the network. In each iteration, the LAs try to form a path in the network such that the link between the starting (s_p) and ending (e_p) nodes of the path belongs to some test link (s_p, e_p). After a path is created, the strength of the path is calculated based on the fuzzy links in the path. Now FLP-DLA assigns the strength of the path to the strength of test link (s_p, e_p), and based on this strength it rewards or penalizes each LA in the path. The main phase is repeated until LAs converge to some paths. Finally, it uses the strength of the test links as the output of link prediction. In order to examine the result of the proposed method, some evaluations are conducted and the results of the proposed method are compared with the ones achieved by other link prediction techniques. In general, the experiments showed that our approach performs better than other strategies.

The rest of this paper is organized as follows. Section II reviews the literature about link prediction problem. Learning automata and distributed learning automata are described in section III. Section IV introduces the proposed DLA-based link prediction method. Section V presents the experimental study for some social network data sets. Section VI summarizes the main conclusion of our research.

II. Link Prediction

A classic definition of the link prediction problem is expressed by: "Given a snapshot of a social network at time t, we seek to accurately predict the edges that will be added to the network during the interval from time *t* to a given future time *t+1*" [2]. The most widespread approach to the problem is to explore the topological/structural patterns of the social network of interest [10]. Different metrics to describe node pairs have already been adopted in previous studies [11] and they explore the structural patterns of the network and commonly provide a degree of proximity/similarity between the nodes. There are many similarity metrics include [2]: 1) Local similarity metrics that only use the local information of a link to calculate similarity metric: Common Neighbors, Salton Index, Jaccard Index, Hub Depressed Index, Hub Promoted Index, Leicht-Holme Newman Index (LHN1), Preferential Attachment Index, Adamic-Adar Index and Resource Allocation Index, 2) Global similarity metrics that can use all information in the network to calculate the similarity metric between two nodes: Katz Index, Leicht-Holme-Newman Index (LHN2), Matrix Forest Index (MFI) and 3) Quasi-local metrics that do not require global topological information but use more information than local indices: Local Path Index, Local Random Walk, Superposed Random Walk, Average Commute Time, Cos+, random walk with restart, SimRank, Resource Allocation index and Local Path index. As previously mentioned, the starting point in these approaches is to extract the values/scores of different metrics that represent the proximity of pairs of nodes. Then, pairs of non-connected nodes are ranked according to a chosen metric (for instance the number of common neighbors) [2]. After that, the top L ranked pairs are assigned as predicted links. To put it another way, it is always assumed that links that have the highest scores are most likely to occur. In the following the seven commonly similarity measures that are used in the experiments are briefly presented:

1) Common Neighborhood: In this measure, two nodes x and y are more likely to have a link if they have many common neighbors. This score is defined as

$$CN(x,y) = |\Gamma(x) \cap \Gamma(y)|$$

where $\Gamma(x)$ denotes neighbors of node x.

2) Salton: This score is defined as

$$Salton(x,y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma(x)| \times |\Gamma(y)|}}$$

3) Jaccard Index [12]: This index was proposed by Jaccard and it is defined as:

$$Jaccard(x,y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

4) Preferential Attachment (PA) [13]: The preferential attachment (PA) algorithm is based on the preferential attachment phenomena rule [14] that is discovered in a variety of social networks. In this method, the link score is set to be the product of the degrees of the involved nodes and it is defined as follows:

$$PA(x,y) = |\Gamma(x)| \times |\Gamma(y)|$$

5) Adamic-Adar Index (AA) [15]: This index is an extension of common neighborhood method such that the less-connected neighbors have more weight and it is defined as:

$$AA(x,y) = \sum\nolimits_{z\in\Gamma(x)\,\cap\Gamma(y)} \frac{1}{\log(\Gamma(z))}$$

6) Katz Index: In this metric, a similarity is defined as the sum of the number of paths with different lengths such that shorter paths have more weights. It is defined by the following equation:

$$Katz(x,y) = \sum\nolimits_{l=1}^{\infty} \beta^{l}.\,|Path(x,y)^{<l>}|$$

Where $|Path(x,y)^{<l>}|$ is the number of paths between x and y with length *l*. It is also shown that the Katz metric can be calculated based on the following equation:

$$Katz = (I - \beta A)^{-1} - I$$

7) LP Index: This index is a restricted version of Katz metric such that only paths in length 1 and 2 is considered. This metric has a lower computational complexity in comparison to Katz and it is defined as the following:

$$LP\ Index(x,y) = A^2 + \epsilon A^3$$

For more information about other similarity metrics please refer to Liben- Nowell and Kleinberg [2].

The node-wise similarity-based approach searches appropriate measures of similarity between two nodes according to the content and/or semantics they present [2]. Each node on the network can be represented as a vector of features. The more similar two nodes are in terms of their particular attributes, the likelier they are to relate. Cosine coefficient, mutual information, and Dice coefficient [2] are examples of techniques used in this approach.

The approaches that are based on probabilistic models try to learn the best probabilistic model that abstracts the network information. The basic idea here is to create the model through a set of parameters θ, given the observed social network G= (V,E) [2]. The existence of a link between the pair of nodes x and y is estimated by the conditional probability P(e(x,y)| θ) [2]. This approach examines the elements of the network with the help of relational data models and enables us to encapsulate relevant information from node relationships. Relational Markov Networks and Relational Bayesian Networks are two examples of models dealt with in this approach.

In the rest of this section, we go on to review recently proposed link prediction methods:

The authors of [16] have proposed a data mining process, called Interaction Prediction, which addresses a particular formulation of the link prediction problem for dynamic networks. Their approach predicts future interactions by combining dynamic social networks analysis, time series forecast, feature selection such as similarity metrics, and network community structure. The proposed method focuses on the links within the communities. The experiments on real-world interaction networks show that the proposed approach achieves encouraging results in both cases of balanced and unbalanced class distribution.

The authors of [17] have provided an approach for predicting future links by applying the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) to optimize weights which are used in a linear combination of sixteen neighborhood and node similarity indices. They have examined a large dynamic social network with over 10^6 nodes. Their method exhibits fast convergence and high levels of precision for the top twenty predicted links.

The authors of [18] proposed a new link prediction using information theory and mutual information of network structure (MI-LP). They compared the model with six typical prediction methods on ten networks. Their results show two good features: improving the link prediction accuracy and the reasonable computational complexity.

In [19] a novel method called Multivariate Time Series Link Prediction is proposed for link prediction in evolving networks. This method integrates (1) temporal evolution of the network; (2) node similarities, and (3) node connectivity information. This method uses existing connections and a calculated similarity metric for each time. The authors finally compare different similarity metrics in their experiments.

In [20] a weighted approach for modeling the occurrence of time is used to generate a different similarity metric and finally, in our previous research, a new link prediction method based on temporal similarity metrics and Continuous Action set Learning Automata (CALA) is proposed [21]. The proposed method takes advantage of using different similarity metrics as well as different time periods. In the proposed algorithm, the link prediction problem is modeled as a noisy optimization problem and a team of CALAs is used to solve the noisy optimization problem. The obtained link prediction results show satisfactory of the proposed method for some social network data sets.

Traditional social network analysis is based on deterministic models, calculating the value of some measures to analyze the relationship that exists between them. However, a simply determined value is not sufficient to describe these relations accurately. Since the fuzzy concept was proposed [22], several studies have shown that the introduction of the fuzzy system, using a fuzzy state to replace the original value, can well solve the problem of uncertainty in social networks. Reference [23] used the fuzzy logic to modify original binary relations into multiple relations among individuals and achieve the same efficiency in social networks. Reference [24] presented a method using the fuzzy logic to explain social relations. Such methods can improve the flexibility of the relationship between social networks, and thus reduce the individual's conflict. The authors of [25] applied a fuzzy model for link prediction based on network characteristics and achieved better results than attained by the traditional method. The introduction of a fuzzy model through fuzzy clustering to predict social network links also shows good performance

[26]. Recently some researchers used the ordered weighted averaging (OWA) operator to obtain the fuzzy relationship between nodes [27]. The method needs some attributes of the network to calculate the relationship. Similarity indices such as common neighbors (CN), Katz, Salton, or Adamic-Adar (AA) can be considered as attributes here. Although the method has the disadvantage of large time complexity, it has advantages of higher prediction accuracy and higher stability. The proposed method uses the fuzzy concept to model the strength of the links in social networks, and it achieves better results in predicting future links in social networks.

III. LEARNING AUTOMATA

A learning automaton [28] is an adaptive decision-making unit that improves its performance by learning how to choose the optimal action from a finite set of allowed actions through repeated interactions with a random environment. The action is chosen at random based on a probability distribution kept over the action-set and at each instant the given action serves as the input to the random environment. The environment responds to the taken action in turn with a reinforcement signal. The action probability vector is updated based on the reinforcement feedback from the environment. The objective of a learning automaton is to find the optimal action from the action-set so that the average penalty received from the environment is minimized. The environment can be described by a triple $E \equiv \{\alpha, \beta, c\}$, where $\alpha \equiv \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ represents the finite set of the inputs, $\beta \equiv \{\beta_1, \beta_2, \ldots, \beta_m\}$ denotes the set of the values that can be taken by the reinforcement signal, and $c \equiv \{c_1, c_2, \ldots, c_m\}$ denotes the set of the penalty probabilities, where the element c_i is associated with the given action α_i. If the penalty probabilities are constant, the random environment is said to be a stationary random environment, and if they vary with time, the environment is called a non-stationary environment. The environments depending on the nature of the reinforcement signal β can be classified into P-model, Q-model and S-model. The environments in which the reinforcement signal can only take two binary values 0 and 1 are referred to as P-model environments. Another class of the environment allows a finite number of values in the interval $[0, 1]$ can be taken by the reinforcement signal. Such an environment is referred to as a Q-model environment. In S-model environments, the reinforcement signal lies in the interval $[a, b]$. The relationship between the learning automaton and its random environment has been shown in Figure. 1 [28]. Learning automata can be classified into two main families [28]: fixed structure learning automata and variable structure learning automata. Variable structure learning automata are represented by a triple $\langle \beta, \alpha, T \rangle$, where β is the set of inputs, α is the set of actions, and T is the learning algorithm. The learning algorithm is a recurrence relation which is used to modify the action probability vector. Various learning algorithms have been proposed. The proposed method uses the following S-model learning: Let α, be the action chosen at step n from the probability distribution p. The linear reward-inaction algorithm is one of the learning schemas and its recurrence equation for updating action probability vector p is defined as the following equation:

$$p_i(n+1) = p_i(n) + a.\big(1-\beta(n)\big).\big(1-p_i(n)\big) - b.\beta(n).p_i(n) \qquad (1)$$

$$p_j(n+1) = p_j(n) + a.\big(1-\beta(n)\big).p_j(n) + \frac{b.\beta(n)}{r-1} - b.\beta(n).p_i(n) \quad j \neq i$$

Where a and b denote the reward and penalty parameters and determine the amount of increases and decreases of the action probabilities, respectively.

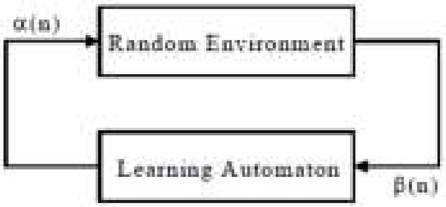

Figure 1. The Relationship between a Learning Automata and Its Random Environment

Learning automata have been found to be useful in systems where incomplete information about the environment, wherein the system operates, exists and recently, several learning automata based approaches have been presented for improving the performance of many applications [29] [30] [31].

1. Distributed Learning Automata

A Distributed learning automata (DLA) [32] [33] shown in Figure. 2 is a network of interconnected learning automata which collectively cooperate to solve a particular problem. Formally, a DLA can be defined by a triple $\langle A, E, T \rangle$, where $A = \{A_1, A_2, \ldots, A_n\}$ is the set of learning automata, $E \subset A \times A$ is the set of the edges in which edge $e(i, j)$ corresponds to the action α_{ij} of the automaton A_i, T is the set of learning schemes with which the learning automata update their action probability vectors. The operation of a DLA can be described as follows. At first, a random node is chosen as the root automaton and it randomly chooses one of its outgoing edges (actions) according to its action probabilities and activates the learning automaton at the other end of the selected edge. The activated automaton also randomly selects an action which results in activation of another automaton. The process of choosing the actions and activating the automata is continued until a leaf automaton (an automaton which interacts with the environment) is reached. The chosen actions, along the path induced by the activated automata between the root and leaf, are applied to the random environment. The environment evaluates the applied actions and emits a reinforcement signal to the DLA. The activated learning automata along the chosen path update their action probability vectors on the basis of the reinforcement signal by using the learning schemes. This procedure repeats until the probability of chosen paths is close enough to unity. For example in Figure. 2, every automaton has two actions. If automaton A_1 selects α_3 from its action set, then it activates automaton A_3. Afterwards, the automaton A_3 will choose one of its possible actions and so on.

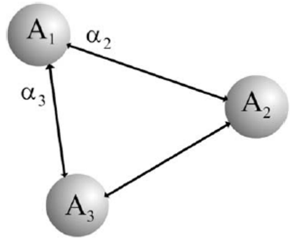

Figure 2. Distributed Learning Automata

IV. Proposed Link Prediction Method In Fuzzy Social Network

This section describes the proposed link prediction method for fuzzy social networks. To do this we first review the fuzzy concepts that we used them in the proposed method. Then, we describe the preprocessing phase to show how the proposed method models one social network into a fuzzy social network. After that, we propose a link prediction algorithm based on the fuzzy social network.

1) Fuzzy Concepts

The fuzzy set theory, introduced by Zadeh [22], is suitable for dealing with the uncertainty and imprecision associated with information concerning various parameters. In this section we briefly review the main concepts and definitions of the fuzzy sets, fuzzy variables and the fuzzy social networks.

Definition 1. A fuzzy set A in R (real line) is defined to be a set of ordered pairs $A = \{(x, \mu_A(x)) \mid x \in R\}$, where $\mu_A(x)$ is called the membership function for the fuzzy sets.

Definition 2. A fuzzy set A is called **normal** if there is at least one point $x \in R$ with $\mu_A(x) = 1$.

Definition 3. A fuzzy set A on R is **convex** if for any $x, y \in R$ and any $\lambda \in [0,1]$, we have $\mu_A(\lambda x + (1-\lambda)y) \geq \min(\mu_A(x), \mu_A(y))$.

Definition 4. A fuzzy **number** is a fuzzy set on the real line that satisfies the conditions of **normality** and **convexity**.

Definition 5. A **L-R** fuzzy number denoted by M = (m, α, λ), has the following membership function:

$$\mu_M(x) = \begin{cases} 0 & x \leq m - \alpha \\ 1 - \dfrac{m-x}{\alpha} & m - \alpha < \mathrm{x} < \mathrm{m} \\ 1 & x = m \\ 1 - \dfrac{x-m}{\lambda} & m < x < m + \lambda \\ 0 & m + \lambda \leq \mathrm{x} \end{cases}$$

Definition 6 [34]. A fuzzy social network is defined as a fuzzy relational structure $\tilde{G} = (V, \tilde{E})$ where $V = \{v_1, v_2, \ldots, v_n\}$ is a non-empty set of actors or nodes and $\tilde{E} = \begin{bmatrix} \widetilde{e_{11}} & \cdots & \widetilde{e_{1n}} \\ \vdots & \ddots & \vdots \\ \widetilde{e_{n1}} & \cdots & \widetilde{e_{nn}} \end{bmatrix}$ is a fuzzy relation on V. Also $\widetilde{\mu_E} = \begin{bmatrix} \mu(\widetilde{e_{11}}) & \cdots & \mu(\widetilde{e_{1n}}) \\ \vdots & \ddots & \vdots \\ \mu(\widetilde{e_{n1}}) & \cdots & \mu(\widetilde{e_{nn}}) \end{bmatrix}$ is a membership function. In other words, each $\widetilde{e_{ij}}$ is a fuzzy variable that shows a relationship between nodes i and j. The proposed method uses the concept of strength of the links as a membership function to transform a social network into a fuzzy social network where each link has a fuzzy number that shows the strength of the link.

The procedure of creating a fuzzy social network will be described in detail in the next section.

2) Preprocessing Phase

This section describes how the FLP-DLA models a static social network into a fuzzy one. The construction of a fuzzy social network based on activity records is done by capturing the activities and interactions among the nodes of the current network. An activity record is defined as a pair: <Date, Co-No>. The Date represents the date of the link occurrence and Co-No represents the number of nodes that are evolving with the corresponding activity. For example in co-authorship social networks, the Co-No is considered as the number of authors that have collaborated in publishing the corresponding paper and in Email networks the Co-No is considered as the number of recipients of an email. An example of an activity record looks like this:

<2002/02/13, 3>

This paper tries to calculate a fuzzy strength for each link based on Date and Co-No variables. To do this it first converts the Date of each activity to a fuzzy one by defining a fuzzy variable for Date variable. Different experts could define fuzzy sets in different ways. It defines Date as a fuzzy variable like the following fuzzy sets: if the used social network is in time interval $[t\ t']$, then it splits this time interval into 3 sections: {very-old, old, recent}. For example if the time interval of the used social network is [1994-2002], the Date fuzzy variable is represented by the fuzzy sets in Figure 3.

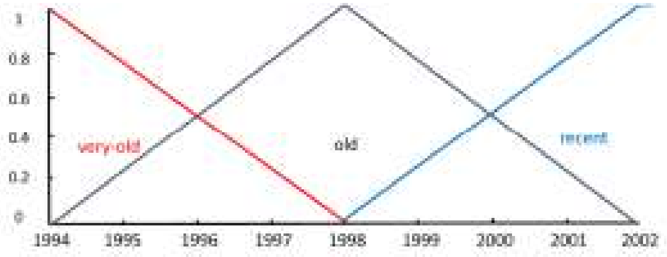

Figure 3. Fuzzy Sets for Date Fuzzy Variable

From Figure. 3, it can be observed that three fuzzy sets have been defined: very-old, old and recent. Each of these fuzzy sets can be represented using a triangular fuzzy number. Table 1 shows the value of the fuzzy sets representing the date variable. The FLP-DLA uses the L-R fuzzy number notation, where a fuzzy number is defined as a tuple (m, α, λ), where m is the mean value, α is the left spread, and λ is the right spread [22].

Table 1. L-R fuzzy numbers for Date variable

Fuzzy Set	L-R Fuzzy Number
Very-old	(1994, 0, 4)
Old	(1998, 4, 4)
Recent	(2002, 4, 0)

Now that we have defined a fuzzy number for each of the possible values assigned to the date variable, we can redefine our social activity records as follows:

<2002/02/13, 3> → <recent, 3> → <(2002,4,0), 3>

Until now, we have introduced an approach to convert Date variable into an L-R fuzzy number. Now we want to use this fuzzy value and Co-No information together to calculate the link strength. We define the link strength as an L-R fuzzy variable, S(n, τ, υ), where n is the mean value, τ is the left spread, υ is the right spread and propose the following equation to obtain the link strength:

$$n = x_m \times e^{-\frac{Co-No}{2}}, \ \tau = x_\alpha \ , \ \upsilon = x_\beta \qquad (2)$$

where x is the Date fuzzy number, x_m is the mean value of the x, x_α is the left spread, x_β is the right spread and the Co-No represents the cooperation number of the link. This function is designed such that the newer papers with lower cooperation number have a larger strength. For example, assume we have a social activity in form of <(2002, 4, 0), 3>. The strength of the link based on Equation 2 is the L-R fuzzy number S(0.2231, 4, 0). So using Equation 2, we finally have a fuzzy social network in which each link has an L-R fuzzy strength. Figure. 4 shows an example of converting a social network to a fuzzy social network. In the next step we use this fuzzy social network to predict future links.

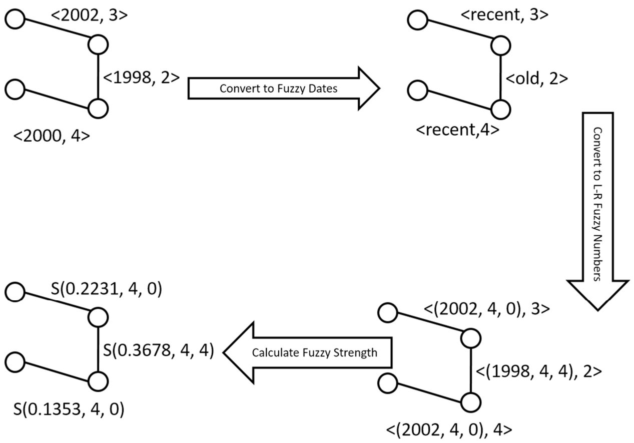

Figure 4. An Example of Converting a Social Network to a Fuzzy Social Network

3) Proposed Fuzzy Link Prediction

This section describes the proposed link prediction based on the fuzzy social network that was introduced and generated in the two past sub-sections. The proposed method takes advantage of DLA to predict future links according to the following procedure: In the beginning, each node of the graph is equipped with a learning automaton, and as a result, a network of learning automata isomorphic to the graph is initially constructed. The set of distributed learning automata is defined by the tuple $< A, \alpha >$, where $A \equiv \{A_1, A_2, \dots, A_n\}$ is the set of learning automata corresponding to the set of nodes, and $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ denotes the set of actions in which $\alpha_i \equiv \{\alpha_i^1, \alpha_i^2, \dots, \alpha_i^{r_i}\}$ is the set of actions that can be taken by learning automaton A_i. In the proposed algorithm, the action set of each learning automaton (e.g. A_i) is to choose one of the neighbors j based on probability p_{ij} that is initialized as:

$$p_{ij} = \frac{1}{d(v_i)} \tag{3}$$

where $d(v_i)$ is the degree of vertex v_i.

At the beginning of the algorithm, all learning automata are considered inactive. The FLP-DLA first chooses a node A_i as the start node from the set of nodes N_{test}. The N_{test} is the set of end-point nodes of links that must be predicted. Automaton A_i is selected and activated. The activated automaton selects one of its neighbors (A_j) according to the probability vector of A_i. Then, A_j selects one of its neighbors A_k according to the probability vector of A_j. This procedure is continued until node A_k belongs to the N_{test} set. So, there is a fuzzy path P_r between A_i and A_k such that the pair (A_i, A_k) belongs to our test set. Now, the FLP-DLA calculates the strength of path P_r, $S(P_r)$, based on the following equation:

$$S(P_r) = \frac{\sum_{l \in links\ in\ P_r} EV(S_l)}{k} \tag{4}$$

Where $EV(S_l)$ is the expected value of link *l's* strength and k is the length of P_r. Since each link's strength is an L-R fuzzy number it calculates the expected value of link *l's* strength, $EV(S_l)$, based on the following equation:

$$EV(x) = n - \frac{\upsilon - \tau}{3} \tag{5}$$

In other words, the strength of path P_r is equal to the average of strength' expected values of the links in the path P_r. Now the FLP-DLA normalizes this strength to a number value in interval [0 1] to generate a β signal and then it rewards or penalizes all the learning automata in path P_r based on the β value using equation (1). Also, it assigns the obtained path strength to the link strength. Then again it chooses a node from set N_{test} and create a fuzzy path. This process is iteratively repeated until a predefined number of iterations are carried out or the process exceeds a predefined threshold of convergence. The threshold of convergence is defined by the product of the action probability vector of learning automata as follows:

$$\tau = \prod_{i \in T} \arg\max\{p_i(t)\} \tag{6}$$

Where t is the iteration number and T is the set of training nodes in the graph. Also $p_i(t)$ shows the probability vector of automata i in iteration t.

At the end of this process, k-top test links with higher strength will be taken as future links. The pseudo code of the proposed algorithm based on the distributed learning automaton is shown in Figure. 5 and Figure. 6. In Figure .5 first a set of learning automata isomorphic to the nodes of the graph is considered and the Date information of the used social network is converted to a set of L-R fuzzy numbers. Then for each link, the fuzzy strength of the link is calculated based on equation (2). In each iteration of the FLP-DLA, a random start node is chosen as the start node of the DLA and the DLA tries to find a path between the endpoints of a test link. When this path is created, the DLA stops and calculates the strength of the obtained path using equation (4) and (5). This strength is used as the reinforcement signal to reward or penalize each LA in the path. Finally, the convergence metric of the DLA is calculated based on equation (6) and until the convergence metric is not reached to a predefined threshold, a random start node is selected again and the procedure of finding the path and rewarding or penalizing the DLA is proceeded again.

```
Algorithm 1. Proposed algorithm for link prediction in fuzzy social networks
Input: Graph <V,E>
Output: The set of predicted links
Assumption:
  Assign an automaton $\boldsymbol{A_i}$ to each vertex $\boldsymbol{v_i}$
  Let $\boldsymbol{\alpha_i} \equiv \{\boldsymbol{\alpha_i^1}, \boldsymbol{\alpha_i^2}, \ldots, \boldsymbol{\alpha_i^{r_i}}\}$ denotes the action set of automata $\boldsymbol{A_i}$
  Let $\boldsymbol{N_{test}}$ be the set of end-point nodes of links that must be predicted
Begin
  Disable all learning automata
  Convert the date information of each link to a fuzzy date variable based on Figure 3.
  Rewrite fuzzy activity set using fuzzy date and Co-No variables.
  Estimate the fuzzy strength of each fuzzy activity based on Equation 2.
  While stopping criteria not met Do
      Choose an automaton randomly as the start automata $\boldsymbol{A_s}$
      Choose a node (action) $\boldsymbol{A_j}$ by $\boldsymbol{A_s}$ according to $\boldsymbol{p_{sj}}$
      Set $\boldsymbol{P_r}$=($\boldsymbol{A_s}, \boldsymbol{A_j}$)
      While ($\boldsymbol{A_s}, \boldsymbol{A_j}$ ) does not belong to the $\boldsymbol{N_{test}}$ Do
          $\boldsymbol{A_s} \leftarrow \boldsymbol{A_j}$
          Choose a node (action) $\boldsymbol{A_j}$ by $\boldsymbol{A_s}$ according to $\boldsymbol{p_{sj}}$
          $\boldsymbol{P_r}$ +=($\boldsymbol{A_s}, \boldsymbol{A_j}$), add the new edge into the current path
      End While
      Calculate the strength of the path $\boldsymbol{P_r}$ based on Equation 4.
      Generate a reinforcement signal $\boldsymbol{\beta}$ based on strength of path $\boldsymbol{P_r}$.
      Reward or penalize each automaton in path $\boldsymbol{P_r}$ using generated $\boldsymbol{\beta}$ and based on Equation 1.
      Set the strength of link (start node of $\boldsymbol{P_r}$, end node of $\boldsymbol{P_r}$) to the strength of $\boldsymbol{P_r}$
  End While
  Return k-top links with higher strength as the predicted links.
```

Figure 5. Pseudo Code of the Proposed Link Prediction in Fuzzy Social Network based on Distributed Learning Automata

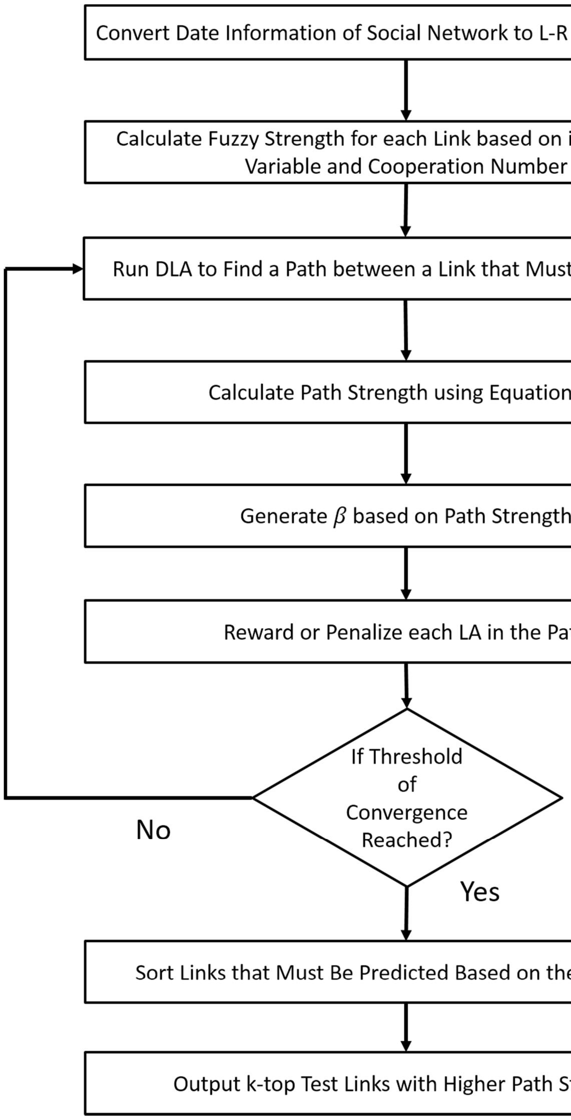
Convert Date Information of Social Network to L-R Fuzzy Numbers
Calculate Fuzzy Strength for each Link based on its Date Fuzzy Variable and Cooperation Number
Run DLA to Find a Path between a Link that Must Be Predicted
Calculate Path Strength using Equation (4)
Generate β based on Path Strength
Reward or Penalize each LA in the Path
If Threshold of Convergence Reached?
No
Yes
Sort Links that Must Be Predicted Based on their Strength
Output k-top Test Links with Higher Path Strength

Figure 5. Block diagram of the Proposed Link Prediction in Fuzzy Social Network based on Distributed Learning Automata

V. EXPERIMENT RESULTS

In this section in order to evaluate the performance of the proposed algorithm, some computer experiments have been conducted and the performance of the proposed algorithm has been compared in term of performance and accuracy. The rest of this section first gives the data set and evaluation metrics that are used in the experiments and then gives a set of experiments.

1. Data Set and Evaluation Metrics

In this section, the social network's data and the evaluation metrics used in our experiments are described. For the experiments developed in this work, we consider the following two groups of networks:

- Co-authorship Networks: A type of social network where the nodes represent the authors and two authors are connected if they have collaborated in a paper. Collaboration network is widely used to understand the topology and dynamics of complex networks. This paper uses three co-authorship networks from three sections of Arxiv[1] and extracted data from the years 1993 to 2003 for all these data sets. The first network is composed by authors that collaborated in theoretical high energy physics[2] (hep-th). The second one is formed by authors who published papers in the high energy physics[3] (hep-ph) and the third one is sampled from collaboration in the Astro Physics[4] (Astro-ph). In these data sets, if an author i co-authored a paper with author

[1] http://www.arxiv.org
[2] http://arxiv.org/archive/hep-th
[3] http://arxiv.org/archive/hep-ph
[4] http://arxiv.org/archive/Astro-ph

j, the graph contains an undirected edge from i to j. If the paper is co-authored by k authors this generates a completely connected (sub) graph on k nodes.

- Email Communication Networks: A type of social network where the nodes of the network are email addresses and if an address i sent at least one email to address j, the graph contains an undirected edge from i to j. In our experiments, we use two email communication data sets: Enron email communication network[5] and Eu-All email communication network[6]. The Enron email communication network includes a data set around half million emails that are public by the Federal Energy Regulatory Commission and we extract data from May 1999 through May 2002 (36 months). Also, the Eu-All email communication network was extracted using email data from a large European research institution and we extracted data from October 2003 to May 2005 (18 months). The network specification of each data set is presented in Table 2. Since these networks are highly sparse, to make computation feasible, we reduce the number of candidate pairs by choosing only the ones that have at least two connections on the network.

In order to do experiments for collaboration networks (Hep-th, Hep-ph and Astro-ph), the proposed method considers the data from 1993 to 2002 as the training data (each year as a time period) and year 2003 as test data. Also for email networks (Enron and EuAll), it considers the first 70% available months as the training data (each month as a time period) and the 30% remaining months as the test data. Also because the FLP-DLA is a stochastic method, all results reported in the next sections are calculated based on the mean of 30 random runs of the FLP-DLA.

Table 2. Network Size in Terms of Nodes and Edges

Data Set	Nodes	Edges	Description
Hep-th	9,877	51,971	Collaboration network of Arxiv High Energy Physics Theory
Hep-ph	12,008	237,010	Collaboration network of Arxiv High Energy Physics
Astro-ph	18,772	396,160	Collaboration network of Arxiv Astro Physics
Email-Enron	36,692	367,662	Email communication network from Enron
Email-EuAll	265,214	420,045	Email network from a EU research institution

Also to evaluate the proposed method with other link prediction methods, we use the two common evaluation metrics as the following:

- AUC Metric [35]: If we rank all of the non-existent links based on their scores, the AUC metric can be interpreted as the probability that a random missing link has a higher score than a random non-existent link. In the algorithmic implementation, at each time it usually picks a missing link and a nonexistent link randomly and compares their scores. if among n independent comparisons, there are n′ times that missing links have a higher score and n" times that they have the same score, the AUC value is:

$$\mathrm{AUC} = \frac{\mathrm{n}' + 0.5\mathrm{n}''}{\mathrm{n}} \qquad (5)$$

  If the AUC value has a value more than 0.5 it is better than the random link prediction algorithm and the farther from 0.5 means the more accurate algorithm.

- Precision [35]: If we predict L links to be connected and L_r links from L links are right, the Precision is defined as:

$$\mathrm{Precision} = \frac{\mathrm{L}_r}{\mathrm{L}} \qquad (6)$$

  Clearly, higher precision means higher prediction accuracy.

2. Experiments

This section presents three experiments to evaluate the performance and accuracy of the proposed method. The first experiment compares the accuracy of the FLP-DLA with the other link prediction methods. The second experiment compares the metrics of

[5] http://www.cs.cmu.edu/~enron/

[6] http://snap.stanford.edu/data/email-EuAll.html

the predicted social network with the original one. Finally, the third experiment compares the computational complexity of the proposed method.

Experiment 1: FLP-DLA Comparison with Other Link Prediction Method

This section compares the proposed FLP-DLA with some classical and recent link prediction methods. In order to improve the comparison a set of algorithms in three categories is chosen:

- Common Similarity Based Link Predictions: in this group of methods the following common similarity based link prediction methods are chosen for comparison with the proposed method: CN, Salton, Jaccard, PA, AA as local similarity metrics, Katz as global similarity metric and LP as quasi-local similarity metric [2].
- Supervised Link Predictions: in this group of algorithms three recent link prediction algorithms, Interaction Prediction (IP) [36], CMA-ES [37] and MI-LP [18] algorithms that try to predict future links are chosen.
- Fuzzy Link Predictions: in this group of algorithms the recent fuzzy link prediction FLP [25] is chosen for comparison with the proposed algorithm.

It should be noted that the parameters of the used algorithms are borrowed from their references. Table 3 and Table 4 present the average AUC and precision scores based on 30 random runs of the FLP-DLA, respectively. Also, in order to have a better comparison, a set of one-tailed t-tests (with α=0.05 and degree of freedom=98) were taken on the results such that if the result of t-test between the FLP-DLA and another algorithm is a negative value then the FLP-DLA statistically outperforms that algorithm. The statistical differences between the experiments obtained with the FLP-DLA and other methods for the test datasets are listed in Table 5. Also, it should be mentioned that for the following tables given in this section, the best results are highlighted.

Table 3 and Table 4 demonstrate that the proposed FLP-DLA is able to achieve an AUC (0.8438) and Precision (0.5965) measures which is significantly better than local similarity-based algorithms: CN (AUC=0.7305, Precision=0.4812), Salton (AUC=0.7103, Precision=0.4603), Jaccard (AUC=0.6290, Precision=0.4098), PA (AUC=0.6173, Precision=0.3889), AA (AUC=0.7130, Precision=0.4635). Given that the proposed fuzzy link prediction uses the global information of the social network to predict future links, it is not surprising that the FLP-DLA performs better than local similarity metrics. The proposed method is also superior to the Katz (AUC=0.8097, Precision=0.5691) and LP (AUC=0.7790, Precision=0.5329) link predictions. Since Katz uses the path lengths between two nodes to predict future links, it shows that the strength of path between two nodes results better prediction than using path length between them. Also in comparison to the LP, because the FLP-DLA is better than Katz it is not surprising that it is better than LP. Also, from the result of Table 3 and Table 4 it can be seen that the FLP-DLA is able to achieve an AUC and Precision which is completely better than IP (AUC=0.8112, Precision=0.5574). Considering both IP and FLP-DLA use time series information, the better result can be obtained from two main reasons: 1) Contrast to IP, FLP-DLA uses non-exact value for the Date of link occurrence and considers the number of collaborations in modelling the link strength and 2) FLP-DLA estimates the path strength using many graph navigations but in IP the next value of every feature in the network is estimated by a forecasting model. From the results, it can be seen that FLP-DLA is better than CMA-ES (AUC=0.8009, Precision=0.5481) because FLP-DLA uses the Date information but CMA-ES doesn't and it only tries to combine similarity metrics by considering a weight for each similarity metric. So because FLP-DLA is completely better than similarity-based methods, it is not out of mind that FLP-DLA would be better than CMA-ES. FLP-DLA is also better than MI-LP (AUC=0.8192, Precision=0.5710) because MLP only uses mutual information of network structure but FLP-DLA uses Date concept and it predicts links using iterative graph navigation. Finally FLP-DLA is small better than FLP (AUC=0.8208, Precision=0.5749) because FLP does not use Date concept and it only uses fuzzy concept in clustering coefficient concept but FLP-DLA uses fuzzy Date concept to include the Date of link occurrence in the predicting task, because we believe that very old links or very weak links are not important for the prediction task. Table 5 also confirms the obtained result and shows that the FLP-DLA outperforms some recent link prediction methods. From the result of this section, it can be concluded that the prediction is better with considering event date instead of static structure and fuzzy strength concept instead of the binary value of link existence.

Table 3. AUC Measures of Proposed FLP-DLA and Other Link Prediction Methods

Method/Data Set	Hep-th	Hep-ph	Astro-ph	Enron	EuAll	Average
CN	0.7945	0.7025	0.6791	0.8123	0.6643	0.7305
Salton	0.7850	0.6854	0.6441	0.8087	0.6285	0.7103
Jaccard	0.6438	0.6026	0.5719	0.7010	0.6259	0.6290
PA	0.6400	0.6101	0.5574	0.6743	0.6049	0.6173
AA	0.7562	0.7109	0.6840	0.8045	0.6097	0.7130
Katz	0.8487	0.8611	0.7486	0.8896	0.7008	0.8097
LP	0.8305	0.8128	0.7115	0.8542	0.6905	0.7790
IP	0.8525	0.8548	0.7328	0.8836	0.7376	0.8112

CMA-ES	0.8462	0.8501	0.7241	0.8601	0.7243	0.8009
MI-LP	0.8950	0.8791	0.7510	0.8547	0.7165	0.8192
FLP	0.8654	0.8752	0.7456	0.8727	0.7452	0.8208
FLP-DLA	**0.9004**	**0.8895**	**0.7560**	**0.9014**	**0.7720**	**0.8438**

Table 4. Precision Measures of Proposed FLP-DLA and Other Link Prediction Methods

Method/Data Set	Hep-th	Hep-ph	Astro-ph	Enron	EuAll	Average
CN	0.5421	0.4532	0.4291	0.5620	0.4196	0.4812
Salton	0.5395	0.4260	0.4021	0.5582	0.3758	0.4603
Jaccard	0.4856	0.3690	0.3620	0.4529	0.3795	0.4098
PA	0.4821	0.3699	0.3052	0.4283	0.3593	0.3889
AA	0.5027	0.4549	0.4503	0.5598	0.3500	0.4635
Katz	0.6135	0.6298	0.5049	0.6473	0.4503	0.5691
LP	0.5703	0.5831	0.4611	0.6184	0.4319	0.5329
IP	0.5853	0.6086	0.5025	0.6204	0.4702	0.5574
CMA-ES	0.6032	0.6140	0.5000	0.6158	0.4637	0.5481
MI-LP	0.6273	0.6382	0.5458	0.5861	0.4578	0.5710
FLP	0.6154	0.6259	0.5249	0.6184	0.4899	0.5749
FLP-DLA	**0.6452**	**0.6401**	**0.5499**	**0.6420**	**0.5057**	**0.5965**

Table 5. The Result of Statistical t-Test between the FLP-DLA and Other Link Prediction Methods

Method/Metric	AUC	Precision
CN	**-36.8581**	**-36.0312**
Salton	**-34.7168**	**-34.8645**
Jaccard	**-87.2769**	**-70.6142**
PA	**-95.4301**	**-65.9421**
AA	**-38.5284**	**-36.7381**
Katz	**-7.5308**	**-5.4301**
LP	**-17.3185**	**-15.5428**
IP	**-6.5712**	**-5.0010**
CMA-ES	**-6.5782**	**-5.6891**
MI-LP	**-3.8194**	**-2.8124**

Experiment 2: Comparison the Metrics of Obtained Network with the Original Network

This experiment compares the topological features of the predicted network which is obtained from prediction result with the original social network. The predicted network is a network with n random predicted edges, where n is the number of edges in the original network. So it has the same number of edges as the original network and the goal in this experiment is to compare the topological features of the original network such as the number of connected components, efficiency, and clustering coefficient with the predicted network. If the values of these features in both the original and predicted network are similar, it can be concluded that the proposed algorithm works well. For each dataset, we test this experiment on its largest connected component. Table 6 summarizes the topological features of the original and predicted network for the largest component of the used data sets. In the Table 6 NUM_{C_x} is the number of the connected components in network X and the size of the largest one. For example, 1222/2 means that this network has 2 connected components and the largest one contains 1222 nodes. In this table, e_x is the efficiency of the network X, C_x is clustering coefficient and K_x is the average degree of the network. From the obtained result it can be seen that the topological features in the original network and the predicted network are approximately similar which confirms the efficiency of the proposed algorithm. Also, from the results of previous section and the reported topological features of the original network, it can be concluded that the proposed algorithm does better in the networks with higher cluster coefficient and higher average degree.

Table 6. The result of topological features in the original network and predicted network

Data Set	NumC$_{Original}$	NumC$_{Predicted}$	E$_{Original}$	E$_{Predicted}$	C$_{Original}$	C$_{Predicted}$	K$_{Original}$	K$_{Predicted}$
Hep-th	4042/1	3465/1	0.185	0.145	0.764	0.652	4.247	3.128
Hep-ph	7031/1	6052/1	0.131	0.117	0.683	0.600	4.021	3.247
Astro-ph	9032/1	7518/1	0.115	0.094	0.581	0.504	2.538	2.065
Email-Enron	14704/1	10874/1	0.073	0.056	0.236	0.201	2.531	1.995
Email-EuAll	18046/1	11582/1	0.054	0.041	0.145	0.94	1.032	0.976

Experiment 3: Computational Complexity

In this experiment, the computation time and the speed of convergence of the FLP-DLA are demonstrated for all five data sets. These algorithms are implemented in MATLAB R2009a on a PC, which has a single CPU of Intel(R) Core i7-960 3.2 GHz and 16 GB of memory. Table 7shows the computation time for proposed algorithm over 2000 iterations and it can be seen that the FLP-DLA requires a rational computation time based on its accuracy.

Table 7. Computation Time Comparison for the Proposed Algorithm (In Seconds)

Method/Data Set	Hep-th	Hep-ph	Astro-ph	Enron	EuAll
CN	115.42	124.58	250.24	2704.20	3187.98
Salton	130.38	132.64	270.42	2930.45	3260.76
Jaccard	125.34	128.59	257.52	2853.87	3305.65
PA	80.42	85.76	176.90	1604.05	1870.65
AA	120.35	124.74	234.51	2801.42	3705.32
Katz	149.87	157.76	310.83	4703.65	5204.61
LP	135.24	151.00	280.09	3487.61	4270.09
FLP-DLA	5121.21	10510.32	18451.87	8100.00	25495.10

VI. Conclusion

One of the challenges of link prediction problem is that many suggested methods do not consider the date of link occurrence and use only the binary version of the social network. In this paper, a new link prediction for fuzzy social network based on distributed learning automata (FLP-DLA) is proposed. The proposed method first models the social network to a fuzzy social network, where each link has a fuzzy strength. The fuzzy strength is defined using the date of link occurrence and the number of collaborations in the corresponding link; because we believe that very old links are not important for the prediction task. Then the fuzzy links are used by distribution learning automata to find the strength of path for any link that must be predicted. DLA tries to find the path strength using reinforcement mechanism and graph navigation. In the proposed method, there is one LA for each node in the network and in each iteration, the DLA tries to generate a path such that the start and end nodes of the path $(\mathbf{s}_p, \mathbf{e}_p)$ belong to some test link. Then it uses the strength of the generated path to find the strength of the link $(\mathbf{s}_p, \mathbf{e}_p)$. It also uses this strength to reward or penalize the set of learning automata that exist along the path. Finally, the links with higher strength are selected as future links.

Our model is compared with some common similarity methods which show that the proposed method is completely better than static similarity based methods because of using date concept and global navigation in the graph. Our method is an iteration-based method, so it takes higher computational time in comparison to the most similarity based methods but its computational time is acceptable based on its accuracy. We also compare the proposed algorithm with some recent link prediction algorithms and here the proposed FLP-DLA is again superior to recent methods in term of accuracy. Finally, we compare the topological features of the obtained network with the original network. The experiment shows that the features of predicted network are similar to the original network that confirms the FLP-DLA. From the result, we can conclude that using the date of link occurrence in terms of fuzzy variables (non-exact value) and the number of collaborations are good metrics to model links strength and also we can conclude that using the links strength to estimate non-existence links strength is a good approach to predict future links.

REFERENCES

1. Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A survey. Physica A: Statistical Mechanics and Its Applications, 390(6), 1150–1170.

2. Al Hasan, M., & Zaki, M. J. (2011). A survey of link prediction in social networks. In Social network data analytics (pp. 243–275). Springer.

3. Garcia Gasulla, Dario, et al. "Social network data analysis for event detection." ECAI 2014: 21st European Conference on Artificial Intelligence: 18-22 august 2014, Prague, Czech Republic: proceedings. IOS Press, 2014.

4. Armengol, E. "Evaluating Link Prediction on Large Graphs." Artificial Intelligence Research and Development: Proceedings of the 18th International Conference of the Catalan Association for Artificial Intelligence. Vol. 277. IOS Press, 2015.

5. Huang, Zan, Xin Li, and Hsinchun Chen. "Link prediction approach to collaborative filtering." Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries. ACM, 2005.

6. Elmagarmid, Ahmed K., Panagiotis G. Ipeirotis, and Vassilios S. Verykios. "Duplicate record detection: A survey." IEEE Transactions on knowledge and data engineering 19.1 (2007): 1-16.

7. Freschi, Valerio. "A graph-based semi-supervised algorithm for protein function prediction from interaction maps." International Conference on Learning and Intelligent Optimization. Springer Berlin Heidelberg, 2009.

8. Liben-Nowell, David, and Jon Kleinberg. "The link-prediction problem for social networks." Journal of the American society for information science and technology 58.7 (2007): 1019-1031.

9. Narendra, K. S., & Thathachar, M. A. L. (2012). Learning automata: an introduction. Courier Corporation.

10. Murata, T., & Moriyasu, S. (2008). Link prediction based on structural properties of online social networks. New Generation Computing, 26(3), 245–257.

11. Al Hasan, M., Chaoji, V., Salem, S., & Zaki, M. (2006). Link prediction using supervised learning. In SDM'06: Workshop on Link Analysis, Counter-terrorism and Security.

12. Gerard Salton and Michael J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, Inc., New York, NY, USA, 1986.

13. Barabási, Albert-László, and Réka Albert. "Emergence of scaling in random networks." science 286.5439 (1999): 509-512.

14. Newman, Mark EJ. "Clustering and preferential attachment in growing networks." Physical Review E 64.2 (2001): 025102.

15. Adamic, Lada A., and Eytan Adar. "Friends and neighbors on the web." Social networks 25.3 (2003): 211-230.

16. Rossetti, G., Guidotti, R., Pennacchioli, D., Pedreschi, D., & Giannotti, F. (2015). Interaction Prediction in Dynamic Networks exploiting Community Discovery. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Ana. s and Mining, 2015, 2015, pp.553-558 : s.n.

17. Bliss, C. A., Frank, M. R., Danforth, C. M., & Dodds, P. S. (2014). An evolutionary algorithm approach to link prediction in dynamic social networks. Journal of Computational Science, 5(5), 750–764.

18. Tan, Fei, Yongxiang Xia, and Boyao Zhu. "Link prediction in complex networks: a mutual information perspective." PloS one 9.9 (2014): e107056.

19. Ozcan, A., & Oguducu, S. G. (2015). Multivariate temporal Link Prediction in evolving social networks. In Computer and Information Science (ICIS), 2015 IEEE/ACIS 14th International Conference on (pp. 185–190).

20. S. Huang, Y. Tang, F. Tang, and J. Li, "Link prediction based on time-varied weight in co-authorship network," in Computer Supported Cooperative Work in Design (CSCWD), Proceedings of the 2014 IEEE 18th International Conference on, 2014, pp. 706–709.

21. Moradabadi, B., & Meybodi, M. R. (2016). Link prediction based on temporal similarity metrics using continuous action set learning automata. Physica A: Statistical Mechanics and Its Applications, 460, 361–373.

22. Zadeh, L. A. (1965). Fuzzy sets. Information and Control, 8(3), 338–353.

23. Nair, P. S., & Sarasamma, S. T. (2007). Data mining through fuzzy social network analysis. In Fuzzy Information Processing Society, 2007. NAFIPS'07. Annual Meeting of the North American (pp. 251–255).

24. Brunelli, M., & Fedrizzi, M. (2009). A fuzzy approach to social network analysis. In Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in (pp. 225–230).

25. Bastani, S., Jafarabad, A. K., & Zarandi, M. H. F. (2013). Fuzzy Models for Link Prediction in Social Networks. International Journal of Intelligent Systems, 28(8), 768–786.

26. Yang, L., Zhang, W., & Chen, Y. (2015). Time-series prediction based on global fuzzy measure in social networks. Frontiers, 1.

27. He, Y., Liu, J. N. K., Hu, Y., & Wang, X. (2015). OWA operator based link prediction ensemble for social network. Expert Systems with Applications, 42(1), 21–50.

28. Thathachar, Mandayam AL, and Pidaparty S. Sastry. Networks of learning automata: Techniques for online stochastic optimization. Springer, 2003.

29. Rezvanian, A., & Meybodi, M. R. (2010). An adaptive mutation operator for artificial immune network using learning automata in dynamic environments. In Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress on (pp. 479–483).

30. Rezvanian, A., & Meybodi, M. R. (2010). LACAIS: learning automata based cooperative artificial immune system for function optimization. In Contemporary Computing (pp. 64–75). Springer.

31. Rezvanian, A., Rahmati, M., & Meybodi, M. R. (2014). Sampling from complex networks using distributed learning automata. Physica A: Statistical Mechanics and Its Applications, 396, 224–234.

32. Beigy, H., & Meybodi, M. R. (2006). Utilizing distributed learning automata to solve stochastic shortest path problems. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 14(05), 591–615.

33. Soleimani-Pouri, M., Rezvanian, A., & Meybodi, M. R. (2012). Solving maximum clique problem in stochastic graphs using learning automata. In Computational Aspects of Social Networks (CASoN), 2012 Fourth International Conference on (pp. 115–119).

34. Hu, R.-J., Li, Q., Zhang, G.-Y., & Ma, W.-C. (2015). Centrality Measures in Directed Fuzzy Social Networks. Fuzzy Information and Engineering, 7(1), 115–128.

35. Lü, Linyuan, and Tao Zhou. "Link prediction in complex networks: A survey." Physica A: Statistical Mechanics and its Applications 390.6 (2011): 1150-1170.

36. G. Rossetti, R. Guidotti, D. Pennacchioli, D. Pedreschi, and F. Giannotti, "Interaction Prediction in Dynamic Networks exploiting Community Discovery," in Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysi. s and Mining, 2015, 2015, pp.553-558 : s.n.

37. C. A. Bliss, M. R. Frank, C. M. Danforth, and P. S. Dodds, "An evolutionary algorithm approach to link prediction in dynamic social networks," J. Comput. Sci., vol. 5, no. 5, pp. 750–764, 2014.