A New Irregular Cellular Learning Automata based Evolutionary Computation for Time Series Link Prediction in Social Networks

Mozhdeh KhaksarManshad
Department of Computer Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran
st_m_khaksarmenshad@azad.ac.ir
Mohammad Reza Meybodi
Department of Computer Engineering, Amirkabir University of Technology, Tehran, Iran
mmeybodi@ aut.ac.ir
Afshin Salajegheh
Department of Computer Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran
a_salajegheh@azad.ac.ir

Abstract. Link prediction, i.e., attempt to predict the event of future associations for a network, is the main task of social network analysis. A great bulk of previous research in link prediction uses the similarity measures of non-connected links at a specific time and a specific goal. For example, new connections can be predicted by ordering links by their similarity scores. A general weakness of most existing researches is that they exclude the present condition of the system and do not take any temporal event into account. In the time-series link prediction issue, timing developments of connection incidences to predict connection chances at particular time needs to be reflected on. In this paper, we propose a new time series link prediction method based on irregular cellular learning automata and evolutionary computation. In the evolutionary procedure suggested here, each vertex (cell) includes a genome as well as a set of learning automata *(LAs)*. The genome represents predicted links for corresponding cell. According to the distributed feature of the recommended approach, each genome is locally developed by local search. An evolutionary computation based on irregular cellular learning automata *(ICLA-EC)* is used to predict connection chances in time period *T+1*. In this method, local information among cells in the progressive periods of time in the network is analyzed to examine the presence of the equivalent connections. Using co-authorship and Email networks to evaluate the suggested approach, it is revealed that consideration of time series connection chances leads to remarkable outcomes.

Keywords: time series link prediction, irregular cellular learning automata and evolutionary computation

1. Introduction

This paper addresses time series link prediction issue via evolutionary computation using a method based on irregular cellular learning automata *(ICLA)*. First, the meaning, usage, and current techniques of link prediction are presented here. Next, the meaning of evolutionary algorithm as well as *ICLA* is presented. At last, the suggested approach called irregular cellular learning automata based evolutionary computation *(ICLA-EC)* is explained in brief.

As the measure of connection between people increments in virtual situations, more helpful social information ends up plainly accessible, filling in as a reason for social network analysis. Link prediction is one of the developing subjects in social network study, which can be explained as the evaluation of a possible connection between two separate individuals in near future [1-3].

Link prediction methods can be classified into two categories--supervised and unsupervised. Similarity scores can be exploited either in a supervised or unsupervised manner. In a supervised one, link prediction

is considered as a classification issue in which diverse scores are considered as predictor symbols by a learning algorithm [4,5]. In an unsupervised manner, a chosen score is plainly ordered so as to predict connections for the finest position groups [6,3,7]. A significant constraint of former investigations is that the similarity scores are usually computed ignoring the network development. The arrangement of social network and its parameters vary after a while, since in most applications they are actually online, non-deterministic and unforeseeable. Consequently, it is not satisfactory to employ deterministic models in practicing actual social network issues. To avoid such an issue, it is recommended to try for link prediction via time series information [8].

According to artificial intelligence, an evolutionary algorithm *(EA)* is a subgroup of evolutionary computation *(EC),* which is a metaheuristic optimization process based on a generic population. Using a population of nominee solutions, evolutionary algorithms try to come up with an appropriate one. This population is assessed and the top solutions are chosen to pair up and form the next generation. Among several generations, the number of proper characteristics augments which leads to a growth in the quality of solutions. An *EA* can be considered as a parallel one. In the majority of parallel evolutionary algorithms *(PEAs),* their duties are classified into different parts which are then solved together using various procedures. Single population fine grained *PEA* is a kind of *PEA* which includes just one population, while contains a spatial construction which restricts the relations among entities [9-11]. In this kind of *PEA*, an entity can just compete and mate with its neighbors. However, the appropriate results may be spread over the whole population, since neighborhood overlap. It is normal to put the entities of a *PEA* into 1, 2, or 3 dimensional grids. It is demonstrated that such a system can be modeled via Cellular Automata *(CAs)* using stochastic rules [9]. In *CA* model which is a non-linear dynamic one, time and space are discrete. Each cell in *CA* model can collect partial information about others and the whole system. Individual cells containing limited information should decide about state changes. Therefore, an uncertainty is entered in decision making. learning in decisions can reduce the mentioned uncertainty [12].

An adaptive decision making algorithm called learning automaton *(LA)* can be used to learn the appropriate action among from a group of authorized ones via interaction with a random environment [13]. In order to enhance the local interactions of *LA* in a complicated systems, *CA* and *LA* algorithms are combined and introduced as a cellular learning automaton *(CLA)* [14]. The *CLA* is superior to *CA* since it attempts to learn optimal actions and is able to enhance the learning process via a combination of *LAs* that interact together. The *CLA* can be divided into synchronous, in which all cells utilize their local rules simultaneously assuming an external clock to cause synchronous events for each cell; and asynchronous in which just some cells are triggered off and the others do not change. The elimination of structural regularity leads to a new model called irregular *CLA (ICLA).*

The combination of the *CLA* model and evolutionary algorithms results in a model called Cellular Learning Automata based Evolutionary Computation (*CLA-EC*) [15]. Such algorithm is able to execute searches in large, complicated and multimodal environments. In *CLA-EC*, as well as other *EAs*, search parameters are explained via genomes. Each genome consists of two elements including model and string genome. A group of *LAs* is called model genome. The second element of genome, string genome, is settled via a set of actions chosen by this group of *LAs*. Applying a local rule, a reinforcement signal vector is generated to feed the group of *LAs*. In line with the learning procedure, the internal state of each *LA* is updated. Therefore, each *LA* of a cell can pick out one of its actions via its probability vector. If the

fitness of the generated string genome is superior to that of the cell, the generated one is chosen as the string genome of the cell. This procedure, i.e. the generation of string genomes by the cells of *CLA-EC* is repeated up to the time that a termination criterion is met.

In this research, an evolutionary computation-based time series link prediction model, called *ICLA-EC-TSLP* (irregular cellular learning automata based evolutionary computation for time series link prediction), is proposed. *ICLA-EC* model is a mixture of *ICLA* [16] and the evolutionary model [15]. For each vertex (cell), the probability of any future connection with a related one is analyzed by each *LA*. Using prior local information of the cell in the network, each *LA* contains two actions to decide about the presence of the equivalent connection at time *T+1*. In accordance with a local rule, a reinforcement signal vector is developed to feed the group of *LAs* residing in the cell. In line with a learning procedure, each *LA* in the cell can update its interior construction using the received signal. The explained stages are repeated up to the time that a termination condition is satisfied. When the algorithm terminates, the genome in each cell represents predicted links for a corresponding cell. In the proposed technique, each link is modeled as a time series and predicted using an optimization tool called *ICLA-EC*. A sequence of time periods *1* through *T* is considered to predict connection incidences in time *T+1*. Therefore, the $ICLA-EC$ progresses along these time periods to examine the presence of equivalent connections. The environments of time *t* form the group of required predictions for time *t+1*, which are computed using local information of each cell at time *t* and neighboring genomes. The main idea by using the *ICLA-EC* in this work is that the *ICLA-EC* utilizes local interaction and learns optimal behavior to adapt to unknown environments. According to empirical achievements, *ICLA-EC-TSLP* surpasses other static link prediction techniques including Jaccard Index (*JI*) [17], Common Neighborhood (*CN*) [3], Adamic-Adar (*AA*) [18], Preferential Attachment (*PA*) [19], quasi-local link prediction methods like local path (*LP*)[20], global link prediction methods like *Katz* [21] and latest link prediction techniques such as learning automata time series link prediction *(LA-TSLP)* [22], Covariance Matrix Adaptive Evolution Strategy *(CMA-ES)* [23],interaction prediction *(IP)* [24], mutual information of network structure link prediction *(MI-LP)* [25], autoregressive integrated moving average link prediction *(ARIMA-LP)* [26], fuzzy link prediction based on distributed learning automata *(FLP-DLA)* [27]both in precision and performance.

The rest of the paper is organized as follows. Section 2 reviews relevant literature on time series link prediction. Section 3 describes *CLA* and *ICLA*. Section 4 introduces a new time series link prediction based *ICLA-EC*. Section 5 presents the experimental study on predicting e-mail and co-authorship links. Section 6 summarizes the main conclusions of the paper and discusses the future direction of our research.

2. Related works

Nearly all previous works accomplish link prediction via static analysis of network data, i.e. no temporal information is used in prediction. Nevertheless, temporal information (such as the minutes during the collaboration of two vertices in the past or the time within the first observation of an association) is an important point that should be analyzed while link prediction [3]. In this section a few previous investigations regarding the exploitation of temporal information for link prediction are discussed in brief.

Considering the idea of temporal events, a new proximity measure for link prediction is introduced by Soares et al. in [28]. In accordance with the creation, preservation or interruption of the correspondence among the nodes in a successive period of time, they introduced a time-related event, in conformity with a

pair of nodes. In their method, an event based score is defined and then modified over time by rewarding the time-related events identified among node pairs and their neighborhood. Via several experiments, they assessed the suggested event based system in diverse situations and analyzed link prediction using co authorship networks.

As another method to organize temporal information, it is possible to consider link prediction like a time series prediction issue [29] and set up a time series for each node pair, in which every series observation is defined as the frequency of connection incidence among nodes within a particular time period. An autoregressive integrated moving average *(ARIMA)* model [30] is used to produce time series predictions and calculate the probability of connection incidence in future. A new link prediction method which utilizes both time series patterns and similarity algorithms is introduced by Huang et al. [26]. According to this algorithm, the time series data is created using link prediction in different periods and its subsequent value is anticipated by *(ARIMA)* model. The eventual predictions are computed according to a mixture of predicted outcomes, besides a selected similarity based method.

Considering temporal similarity and continues action set learning automata *(CALA),* Moradabadi et al. [31] introduced a new algorithm which uses various similarity metrics and time periods. In this method, the link prediction issue is considered as a noisy optimization one and a group of *CALA* is applied to find the solution to this problem. The acquired link prediction outcomes demonstrate that the suggested algorithm is appropriate for some social network datasets. Similarly, another time series link prediction algorithm is proposed in [22], which utilizes learning automata to predict the existence or non-existence of every connection at time $T+1$ via some similarity metrics from time *1 to T*. Also in [27] authors proposed link prediction method based on distributed learning automata (*DLA*) and fuzzy concepts (*FLP-DLA*). *FLP-DLA* estimates the strength of test links by using information of the network. They used the strength of test link as the output of the link prediction method.

For link prediction in evolving networks, a new algorithm called Multivariate Time Series Link Prediction is introduced by Özcan et al. [32], which combines (1) temporal evolution of network, (2) node similarities, and (3) node connectivity information. In this algorithm, available connections and a computed similarity metric are used for each time. The authors performed several experiments to compare various similarity metrics.

In [23], the authors proposed a method to predict future connections employing the Covariance Matrix Adaptive Evolution Strategy (*CMA-ES*) to enhance the weights used in a linear mixture of sixteen neighborhood and node similarity indices. They analyzed an enormous dynamic social network with more than 10^6 nodes. Their proposed approach demonstrates rapid convergence and top accuracy for the first twenty predicted connections.

In[24] authors proposed a new link prediction problem for dynamic networks that called interaction prediction (*IP*). The proposed method used feature selection and time series forecast to predict new links in near future.

Employing information theory and mutual information of network structure (*MI-LP*), the authors of [25] introduced a novel link prediction approach. They used ten networks to analyze their method and compare it with six common prediction algorithms. The achieved outcomes demonstrate an improvement in link prediction precision and a logical computational complexity.

3. Background

In this section, we briefly review problem formulation, learning automata and cellular learning automata.

3.1. Problem Formulation

In this paper undirected unipartite graphs are studied entirely. Following [26,28], the time series link prediction problem is officially introduced. At first, our temporal network dataset is separated into time sliced snapshots from time 1 to T, each of them demonstrates the state of network at various time periods. Considering V as a set of vertices (cells), $V = (1,2,\dots,n)$. A graph series is defined as a list of graphs $(G_1, G_2, \dots, G_T)$ coinciding with a group of symmetric adjacency matrices $(M_1, M_2, \dots, M_T)$. At time points t and t+1, the graphs include equal set of nodes, yet various set of connections. In exchange for edges in $E_t(i,j)$, there is an $N \times N$ matrix M_t with nonzero elements $M_t(i,j)$. The value of $M_t(i,j)$ is chosen among the set $\{0,1\}$ and is used to analyze the existence or non-existence of edge (i,j) within time period (t). Considering such a graph, the goal of the proposed link prediction method is to predict the incidence probabilities of edges at time $T+1$.

3.2. Learning Automata

A learning automaton (*LA*) [13,33] is an adaptive decision-making tool that learns the optimal action through repeated interactions with an unknown random environment. This process is done by interacting with the automaton and the random environment. At each stage, the *LA*, based on a probability distribution of the action-set, chooses an action out of a set of finite actions and applies that to the environment.

After that, the environment evaluates the selected action and, a feedback is received from the environment by the automaton which is used to update the probabilities of actions [33]. Fig. 1 indicates how an automaton interacts with its environment.

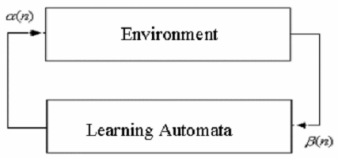

Figure1. The interaction between learning automata and environment

Subsequent to the automaton choosing the action, the reinforcement signal will be received from the environment. When the positive response ($\beta = 0$) is received, the action probabilities are updated through Equation (1):

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)] \qquad (1)$$
$$p_{j \neq i}(n+1) = (1-a)p_j(n)$$

When the negative response ($\beta = 1$) is received from the environment, action probabilities are updated through Equation (2):

$$p_i(n+1) = (1-b)p_i(n) \qquad (2)$$
$$p_{j \neq i}(n+1) = \left(b/r - 1\right) + (1-b)p_j(n)$$

Where a and b are the reward and penalty parameters. When $a = b$, it is called L_{RP}. If $b = 0$, it is called L_{RI} and if $0 < b << a < 1$, it is called $L_{R\varepsilon P}$.

3.3. Cellular Learning Automata and Irregular Cellular Learning Automata

Cellular learning automaton (*CLA*) [14], which is made by combining *CA* [34] and *LA*, is a capable scientific model for some decentralized issues. During the interaction of multiple automata together, *LA* potentials can be recognized completely. When each cell in *CA* is supported by an *LA*, a *CLA* is created. The action probability of the LA living in a cell is used to identify its state. Similar to *CA*, there is a rule that controls the performance of *CLA*. The reinforcement signal to the *LA* which exists in a cell, is identified by the rule for *CLA* and the actions chosen by adjacent *LAs*.

The generalized form of conventional *CLA* called irregular cellular learning automata *(ICLA)* [16] can get over the constraints of a rectangular grid construction. In an undirected graph such as *ICLA*, each node is actually a cell supported by an *LA*, in which the adjacent nodes of any specific node form the local environments of that cell. Regardless of its irregular construction, the performance of *ICLA* is identical to that of *CLA*.

4. The Proposed Irregular Cellular Learning Automata based Evolutionary Computation for Time Series Link Prediction (*ICLA-EC-TSLP*)

In this section, we propose a new evolutionary algorithm called irregular cellular learning automata based evolutionary computation (*ICLA-EC*). *ICLA-EC* is an application of synchronous *CLA* in which the restriction of the regular structure is overcome (Fig.2). The proposed algorithm is a hybrid algorithm based on the *ICLA* and *EC*.

In this method there is a connection between genomes and vertices (cells) of the network. Genomes of all cells are used to form the population. In the beginning, all LAs in the *ICLA-EC* select their actions simultaneously. These actions which are selected from a vertex identify the genome of that vertex. According to a local rule, reinforcement signals are used to create a vector which is then delivered to the *LAs* existing in vertices. In accordance with a learning procedure, all LAs in the *ICLA-EC* modify their action probability vectors. This procedure is reproduced as late as some preset conditions are obtained.

Formally an irregular cellular learning automata based evolutionary computation with n vertices (cells) and m *LAs* in each cell is a structure $\mathcal{A} = (G < V, E >, A, N, \Phi, \mathcal{F})$, where

1. G is an undirected graph, with V as the set of vertices (cells) and E as the set of edges (adjacency relations).
2. A is the set of LAs assigned to *ICLA-EC* , where A_i is the set of *LAs* assigned to cell c_i.
3. Φ is a finite set of actions.
4. Neighborhood vector of any particular cell c_i, is defined as the set of all *LAs* residing in the adjacent cells of the cell c_i, that is $N_i = \{A_j | \{i,j\} \in E\}$.
5. $\mathcal{F}: \Phi \to \beta$ is the local rule of the *ICLA-EC*, where the set of states of all neighbors of particular cell c_i is defined as $\varphi_i = \{\varphi_j | \{i,j\} \in E\} \cup \{\varphi_i\}$ and β is the set of values that the reinforcement signal can take. The local rule computes the reinforcement signal for each *LA* based on the current actions selected by the neighboring *LAs*.

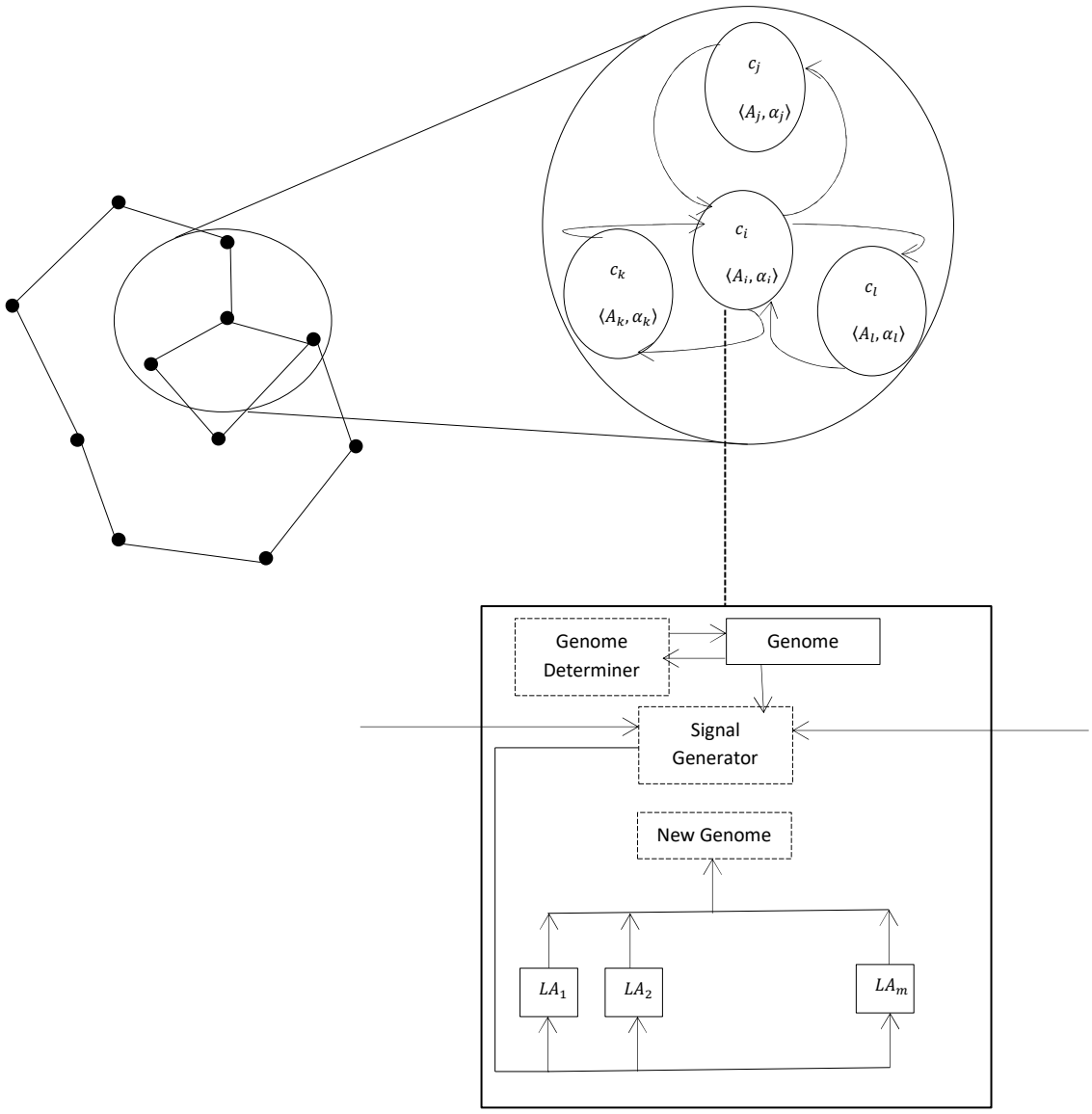

Figure 2. Irregular Cellular Learning Automata Based Evolutionary Computation

In what follows, we present an algorithm based on *ICLA-EC* for time series link prediction (*ICLA-EC-TSLP*). Each cell c_i of the network can be modeled by a tuple $< \alpha_i, A_i >$ where a vector α_i called genome of the cell which represents the predicted links of cell c_i, a vector A_i called model genome and each *LA* in A_i representing a test link that corresponds to cell c_i in time *T+1*. The subgraph of *G* containing only the edges observed at time t is called one time period.

ICLA-EC-TSLP employs time periods 1 through T consecutively to predict future connections in time period $T + 1$. When prediction operation of time period t is completed, the *ICLA-EC-TSLP* proceeds to the next time period while the neighborhood vectors of all cells are modified. Then, in the next time period, every cell attempts to enhance its estimation via the novel environment $t + 1$ and predict the connection incidence of the equivalent link for time period $t + 2$ and the rest. In general, *ICLA-EC-TSLP* contains several principal stages. In each cell there is one *LA* for each node that might be linked to

corresponding cell. Each *LA* includes a set of two actions {0,1}. At first, the probability of each of the two actions is set to 0.5. The action 0 demonstrates that the equivalent test connection should not turn up in prediction outcome while the action 1 shows that the equivalent test connection should be present in the prediction result. In every time period, the steps bellow will be reproduced up to the time a termination condition is satisfied. Considering that the groups of cells run in parallel, the life cycle of a given cell is explained as follows.

1. All *LAs* choose their actions using their action probability vector.
2. Concatenate the chosen actions of *LAs* and generate a new genome α.
3. The fitness of genome is computed. If the fitness of the new genome is better than the previous one, it will be replaced.
4. A set of $N_s(i)$ the neighboring cells of each cell c_i with fitness greater than threshold γ are selected for mating. Note that the mating in this context is not reciprocal, i.e., a cell selects another cell for mating but not necessarily vice versa.
5. The number of cells with the same value of genes is counted. Let cell $N_{ij}(k)$ be the number of cell jth genes that have the same value of k at the selected neighboring cells of c_i(exploration phase). Then, the reinforcement signal for jth *LA* of cell c_i is computed as[15]:

$$\beta_{ij} = \begin{cases} U[N_{ij}(1) - N_{ij}(0)], & if\ \alpha_i^j = 0 \\ U[N_{ij}(0) - N_{ij}(1)], & if\ \alpha_i^j = 1 \end{cases} \qquad (3)$$

Where α_i^j is the value of the jth gene in the ith genome and $U(.)$ is the step function. This process is described in Fig.3.

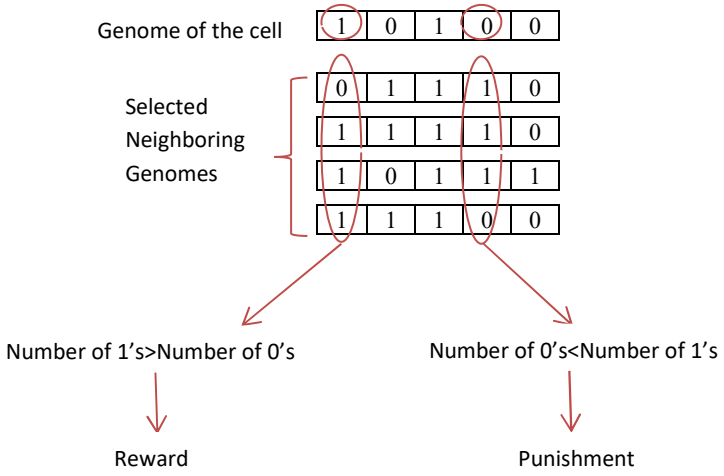

Figure3. The process of computing the reinforcement signal for each *LA*

Then, the ultimate prediction outcome is calculated in another step. When the actions of *LAs* for a genome existing in a cell are selected, its fitness value is computed via Equation (4).

$$fitness\ of\ string\ genome = \frac{\#correct\ link}{\#predicted\ link} \qquad (4)$$

The environment of time period t is a group of predictions for time period $t+1$. Therefore, the time series information of time t and neighboring genomes are employed by *ICLA-EC* to create prediction for

the next time $t+1$. In the following repetition, a novel action is chosen by every *LA* in vector A, using a new action's probability distribution. This process continues until a termination criterion is met.

Then the *ICLA-EC-TSLP* goes to time period $t+1$ and starts to learn the link existence using the current action's probability distribution in the new environment $t+1$. The pseudo-code of *ICLA-EC-TSLP* can be found in algorithm 1.

The suggested *ICLA-EC-TSLP* method includes three steps: *ICLA* modeling, local search, and prediction. In the following section, the principal steps of the suggested algorithm are explained completely.

4.1.1. Modeling Phase

In this stage, a model based on *ICLA-EC* is created to solve time series link prediction problems in social networks. Like other evolutionary algorithms, in *ICLA-EC*, the parameters of search space are explained in the form of genomes. Therefore, each cell contains two components, a group of *LAs* and a genome. The set of actions chosen by this group of *LAs* establishes the genome. In each cell c_i, there is a *LA* in exchange for every test connection which corresponds to the cell c_i in time $T+1$ and should be predicted.

4.1.2. Local search Phase

There is a series of time periods $\{t_1, t_2, \dots, t_T\}$. We define the environment of time period t as a set of predictions for time period $t+1$ that is calculated based on the local information of cell c_i of time t and based on its genome and its adjacent genomes. To execute link prediction, *ICLA-EC* commences its learning procedure in time period one. The action set of a *LA* is a group of actions $\{0,1\}$ that controls the existence or non-existence of the equivalent connection respectively. Every *LA* in a cell contains a probability distribution p_0, p_1 which points out the probability of selecting 0 or 1, respectively. In time period one, every *LA* in cell c_i selects one of its actions via its action probability vector and employs it to control the presence of connection for time 2. Since no previous information exists about the presence of the connection, for each *LA* in a cell the probability of selecting action 1 (connection existence) is set to 0.5, i.e. $p_1 = 0.5$. Therefore, in *ICLA-EC-TSLP* model, in the action selection phase, every *LA* of a cell selects action 1 and 0 according to p_1 and $1-p_1$, respectively. Using Equation (4) the fitness of every genome of the cell is computed. Consequently, if the fitness of this genome is greater than that of the cell, the new genome replaces the genome of the cell. During this step, just the action probability vector is altered by local search, not the value of genome. Hence, during the first step, in exchange for each cell c_i, a number of its neighboring genomes are chosen according to their fitness. Then based on the selected genome and equation(3), a reinforcement signal is generated and the distribution probability of each *LA* in cell c_i is updated.

4.1.3. Prediction Phase

For every *LA* in a cell, the action with an upper probability is selected as the ultimate prediction outcome for the related connection. In other words, in time period $T+1$, for every test connection LA_i^j in a cell c_i, the eventual prediction is explained as p_i^j, and computed according to the following rule:

$$p_i^j = \begin{cases} \alpha_1 & p_1 > p_0 \\ \alpha_0 & else \end{cases} \quad (5)$$

At last, the *ICLA-EC-TSLP* algorithm considers set P values as the result of the suggested link prediction method.

Algorithm 1: Pseodo code of the proposed time series link prediction based on ICLA-EC

Let $M_1, M_2, \dots, M_T$ be the adjacent matrix for time 1 through T.
Let G be the maximum number of generations for one time period.
Let g, t be the generation counter and time period counter and initially set to 0 and let T be the total number of time periods.
Let α in each cell, be the set of LA indexes with 1 through j, in each cell one learning automaton for each link that must be predicted for that cell.
Set the initial probability distribution of choosing action 1 for each learning automaton j in genome of each cell to be $p_1(j) = 0.5$.
While $t < T$ do
  While $g < G$ do
    For each cell c_i in ICLA-EC do in parallel
      Select actions $\{\alpha_i^1, \alpha_i^2, \dots, \alpha_i^j\}$ of LAs in cell c_i based on the action probability distribution of the LAs.
      Evaluate the new $genome$ based Equation (4).
      If fitness($new\ genome$)>fitness($old\ genome$) then
        Accept the $new\ genome$
      End if
      Select a number of genomes from neighbors of cell c_i based on its fitness
      Generate the reinforcement signal based on the selected genomes and the environment of the time period t and based on its $genome$ and $adjacent\ genome$s of selected cells
      Using Equation (1),(2)Update the probability distribution of LAs of cell c_i.
    End parallel for
  $g = g + 1$
  End while
$t = t + 1$
End while
For each generation
  For each test link of genome
    $\{p_i^1, p_i^2, \dots, p_i^j\}$ be the output of prediction result based on Equation (5).

5. Experiments and Results

In this section, the experimental evaluation of the suggested method, as well as the achieved results are presented. In section 5.1 the required social network data along with the analysis process is explained. In section 5.2 the required assessment metrics for our investigations, together with two proposed experiments are described. we've Compared the suggested method with link prediction algorithms in section 5.3. Finally, the precision rate of *ICLA-EC-TSLP* is analyzed in section 5.3.1.

The introduced *ICLA-EC-LP* is compared with two categories:

1. Topology based similarity metrics: in this group we try to compare the similarity of two nodes by using the structure of the network. Of this set of algorithms we chose local similarity metrics that only use the local information of a link to calculate similarity metric such as *CN* [3],*JC* [17],*PA* [19],*AA* [18], global similarity metrics that can use all information in the network to calculate the similarity metric between two nodes such as *Katz* index [21], quasi-local metrics that do not

require global topological information but use more information than local indices such as *LP* [20].

2. Learning based metrics: based on the features provided by basic link prediction metrics, internal attributes, and external information, many learning based link prediction methods are proposed in recent years. In this group we chose feature based classification algorithms such as *IP* [24], *CMA-ES* [23], *MI-LP* [25], *ARIMA-LP* [26] and learning automata based algorithm like *LA-TSLP* [22], *CALA* [31] and *FLP-DLA* [27].

We have compared *AUC* of the *ICLA-EC-TSLP* with local similarity metrics on *Astro-ph*, *Hep-ph*, *Hep-th*, *Email Enron*, *College MSG* Datasets in tables 3, 4, 5. We have compared *AUC* of our algorithm with some recent approaches Katz as global similarity metric, LP as quasi-local metric and *IP*, *CMA-ES*, *MI-LP*, *LA-TSLP, ARIMA-LP, CALA, and FLP-DLA* as learning based metrics in table 6.

5.1. Data and Setting

With a specific end goal to assess the execution of the proposed method, data from three co-authorship networks, one College message network, and one email network were used in our experiments.

A vertex in a co-authorship network corresponds to an author while an edge demonstrates that two particular nodes have co-authored a paper [35] . In other words, in the co-authorship network used in this section, the publishing year of the co-authored paper is saved by every edge. The required co-authorship network data to analyze the proposed algorithm was gathered using arXiv[1] e-print, which preserves an immense database of various scientific articles. To analyze the proposed approach, three co-authorship networks are considered and for all these datasets the required data is gathered for the interval between 1993 and 2003. The first network includes papers of authors who went along theoretical high energy physics (*Hep-th*) while the second one contains the papers of those who collaborated in high energy physics (*Hep-ph*) Finally, the third network involves papers of authors who studied Astro Physics area (*Astro-ph*). Co-authorship networks are really scattered, therefore, in order to improve calculation process, it is necessary to lower the number of candidate pairs. Therefore, just those with not less than two contributions between 1993 and 2003 are selected.

According to United States' Federal Energy Regulatory Commission, Enron email communication network is a public dataset and is used as the second dataset to analyze the proposed approach. The required data is gathered for the period between May 1999 and May 2002 (36 months). In the Enron email network, a node corresponds to an email address, while an edge demonstrates that address j has received at least one email from address i [36].

A College messaging network is considered as the third dataset. The users of this network are pupils of California University, Irvine. The goal of this society is to help pupils extend the group of their friends via fortification of social relations among them. This network dataset includes an interval between April and October 2004 [37].

In order to do experiments on Hep-th, Hep-ph, and Astro-ph, we consider the data from 1993 to 2002 as the training data (each year as a time period) and year 2003 as test data. For Enron email network, we

[1] Arxiv.org eprint archive

consider each three months as a time period. Also for the College message network, we consider each month as a time period.

Table 1. Network Size in Terms of Nodes and Edges

Data Set	Nodes	Edges	Description
Hep-th	9,877	51,971	Collaboration network of ArXiv's High Energy Physics Theory
Hep-ph	12,008	237,010	Collaboration network of ArXiv's High Energy Physics
Astro-ph	18,772	396,160	Collaboration network of ArXiv's Astro Physics
Email-Enron	87,273	1,148,072	Email communication network from Enron
CollegeMsg	1,899	59,835	College message network of students at the University of California

5.2. Evaluation Metrics

We use the two common evaluation metrics to evaluate the proposed method with other link prediction methods as follows [3]:

***AUC*:** Considering the score given to all non-observed links, the *AUC* value can be interpreted as the probability that a randomly chosen missing link is given a higher score than a randomly chosen non-existent link. We randomly pick a missing link and a non-existent link to compare their scores. If among n independent comparisons, there are n' times the missing link having a higher score and n'' times they having the same score, the AUC value is:

$$AUC = \frac{n' + 0.5n''}{n} \qquad (6)$$

Precision: Given the rankings of the non-observed links, precision is defined as the ratio of relevant items selected to the number of items selected. That is to say, if we take the top-L links as the predicted ones, out of which $L_r$links are right, then precision equals Equation (7). Clearly, higher precision means higher prediction accuracy.

$$Precision = \frac{L_r}{L} \qquad (7)$$

5.3. Link Prediction Comparison

In this section we provide results for *ICLA-EC-TSLP* and compare them with those of other similarity based approaches. Each quantity equals to the average of 20 operations. According to experimental outcomes, in the conducted analysis, *ICLA-EC* is used with reward parameter $a = 0.04$ and penalty parameter $b = 0.01$. The convergence of all learning automata is considered as a termination condition in a specific time period. The algorithm may also terminate after 2000 iterations in a specific time period.

Table 2. The Four Similarity based methods. Let Γ(x) Denote Neighbors of the Node x, The Degree of Node x is Represented by k_x.

Topological similarity indices	Formula	Description
Common Neighborhood[3]	$CN(x,y) = \lvert\Gamma(x) \cap \Gamma(y)\rvert$	Two nodes x and y, are more likely to have a link if they have many common neighbors.
Jaccard Index[17]	$Jaccard(x,y) = \frac{\lvert\Gamma(x) \cap \Gamma(y)\rvert}{\lvert\Gamma(x) \cup \Gamma(y)\rvert}$	Measures the probability that a neighbor of x or y is a neighbor of both x and y.
Preferential Attachment[19]	$PA(x,y) = k_x \times k_y$	Gives higher scores to pairs of nodes for which one or both have high degree.
Adamic-Adar Index[18]	$AA(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log(\lvert\Gamma(z)\rvert)}$	This index refines the simple counting of common neighbors by assigning the less-connected neighbors more weight.
LP Index[20]	$LP\ Index(x,y) = A^2 + \varepsilon A^3$	a restricted version of the Katz metric such that only paths of length 1 and 2 are considered.
Katz Index[21]	$Katz(x,y) = \sum_{l=1}^{\infty} \beta^l . \lvert path(x,y)^{<l>} \rvert$	Sum of the number of paths with different lengths, such that shorter paths have more weights.

5.3.1. Co-authorship Networks

In this experiment, we consider each year as a time period. To predict a test link in year α we use an *ICLA-EC* which consists of the first time period through T. To do this, we consider 10 years from 1993 to 2002, and predict the test link for the years 1996 through 2003. The link prediction results of the proposed algorithm *(ICLA-EC-TSLP)* for years 1996 through 2003 are compared to some similarity-based link predictions using the *AUC* metric. Table 3 show *AUC* value of the proposed algorithm and similarity-based method for *Hep-ph*, *Hep-th*, and *Astro-ph* datasets. The results show that the proposed method is able to achieve a better *AUC* measure than *CN*, *PA*, *AA*, and *JI*.

Table 3. Comparison of *AUC* of the *ICLA-EC_TLSP* with Similarity-based Methods on *Astro-ph, Hep-ph, Hep-th* Datasets

Method	CN			JC			AA			PA			ICLA-EC-TSLP		
Year	Astro-ph	Hep-ph	Hep-th	Astro-ph	Hep-ph	Hep-th	Astro-ph	Hep-ph	Hep-th	Astro-ph	Hep-ph	Hep-th	Astro-ph	Hep-ph	Hep-th
1996	0.5240	0.5230	0.5350	0.5243	0.5255	0.5347	0.5247	0.5250	0.5378	0.5273	0.5125	0.5150	0.8590	0.8247	0.7863
1997	0.5233	0.5705	0.5480	0.5287	0.5725	0.5540	0.5190	0.5735	0.5494	0.5110	0.5675	0.4910	0.8920	0.8069	0.7924
1998	0.5507	0.5250	0.5390	0.5577	0.5265	0.5413	0.5647	0.5205	0.5443	0.5090	0.5125	0.5005	0.8802	0.8926	0.7863
1999	0.5737	0.5580	0.5677	0.5850	0.5530	0.5630	0.5950	0.5545	0.5610	0.5263	0.5278	0.5114	0.8821	0.8190	0.7582
2000	0.5503	0.5505	0.5443	0.5600	0.5465	0.5507	0.5525	0.5470	0.5486	0.5112	0.5315	0.5263	0.8577	0.8265	0.7916
2001	0.5583	0.5365	0.5507	0.5365	0.5415	0.5510	0.5750	0.5460	0.5590	0.5167	0.5015	0.5312	0.8467	0.8729	0.7835
2002	0.5250	0.5840	0.5600	0.5437	0.5805	0.5593	0.5937	0.5850	0.5587	0.5214	0.5412	0.5175	0.8550	0.8544	0.74
2003	0.5350	0.5275	0.5577	0.5550	0.5385	0.5577	0.5788	0.5420	0.5537	0.5188	0.5510	0.5212	0.8650	0.8515	0.7849

Using co-authorship datasets, the growing precision diagram of *ICLA-EC-TSLP* is demonstrated in Fig. 4. According to achieved results, during the first periods of time *ICLA-EC-TSLP* acts casually, while after several time periods it learns a new pattern which is the combination of older ones as well as novel environmental patterns.

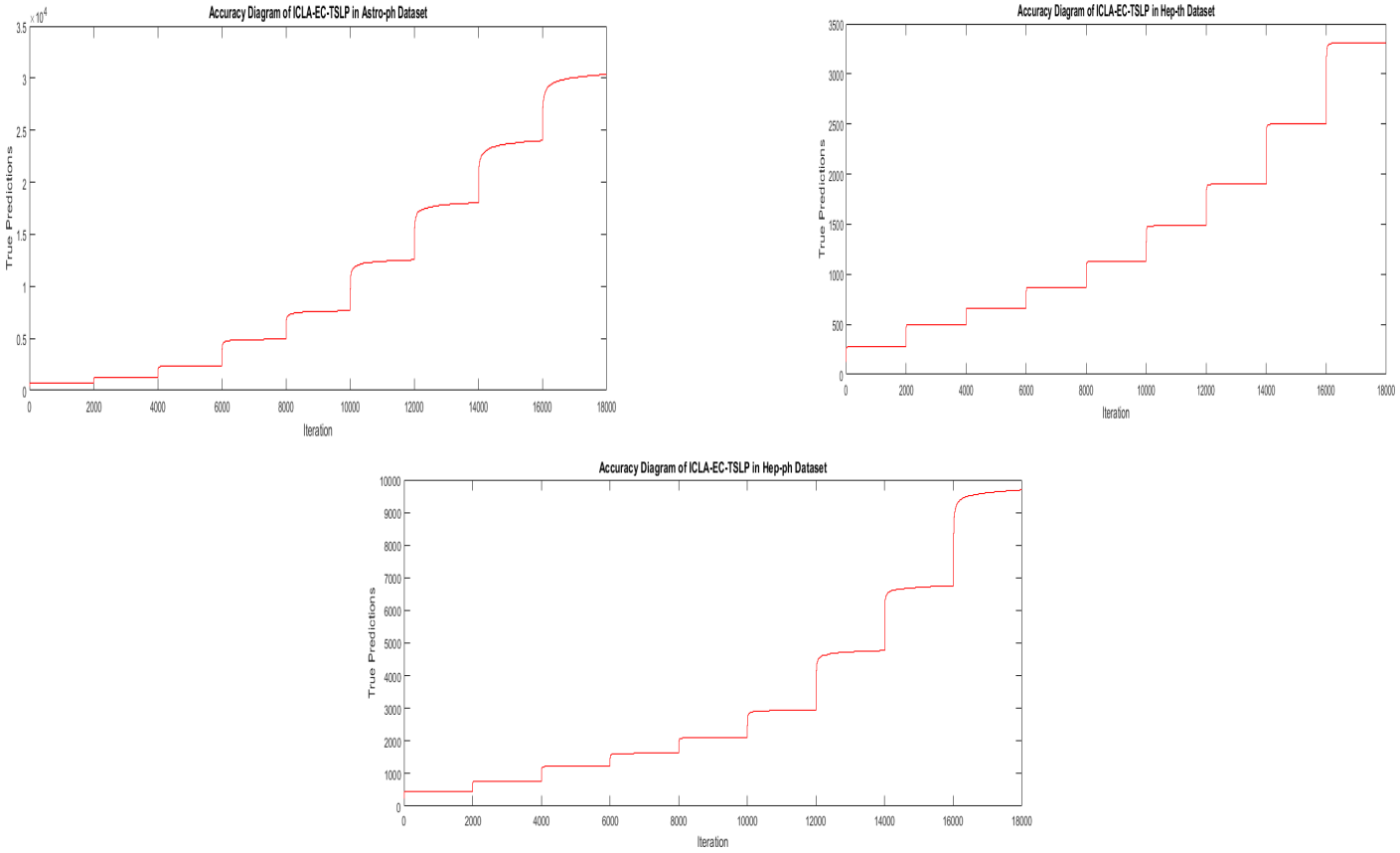

Figure 4. The progressive accuracy diagram of the *ICLA-EC-TSLP* for co-authorship datasets

5.3.2. Enron E-mail Dataset

In this dataset, we consider each three-month as one time period. To do this, we consider 6 time periods from January 2000 to December 2001, and predict the test link for the time periods May 2000 through April 2002 and compare it with some similarity-based link prediction using the *AUC* metric.

Table 4. Comparison of *AUC* of the *ICLA-EC_TLSP* with Similarity-based Methods on Enron E-mail Dataset

Method Year/month	CN	JC	PA	AA	ICLA-EC_TSLP
2000-5to8	0.6600	0.6627	0.6561	0.6593	0.7014
2000-9to12	0.6584	0.6565	0.5789	0.6578	0.6112
2001-1to4	0.6531	0.6548	0.5083	0.6496	0.6694
2001-5to8	0.5899	0.5835	0.5928	0.5919	0.6247
2001-9to12	0.6194	0.6209	0.5854	0.6155	0.6459
2002-1to4	0.5911	0.5916	0.5811	0.5841	0.7212

The results reported in Table 4 also show that the proposed time series link prediction method is able to achieve a better measure than *CN, JI, PA,* and *AA* on average.

5.3.3. *CollegeMsg* Network

In this experiment, we compare *ICLA-EC-TSLP* with some similarity-based methods using a College messaging network. To do this, we consider one time period of each training month and predict the links of the test months and compare it with some similarity-based link prediction method using the *AUC* metric. Our analysis includes all users that sent or received at least one message during that period. The obtained results of the proposed algorithm and its comparison with other methods for College messaging networks are reported in Table 5.

Table 5. Comparison of *AUC* of the *ICLA-EC_TLSP* with Similarity-based Methods on *CollegeMsg* Dataset

Method Year/month	CN	JC	PA	AA	ICLA-EC_TSLP
2004-6	0.4999	0.5330	0.4883	0.3969	0.6963
2004-7	0.5330	0.5180	0.5129	0.4704	0.7849
2004-8	0.5024	0.4989	0.4871	0.4276	0.8131
2004-9	0.4904	0.4871	0.4734	0.3796	0.7788
2004-10	0.4451	0.4414	0.4291	0.3802	0.7825

5.3.4. *ICLA-EC-TSLP* comparison with recent link prediction methods

In this section, the suggested *ICLA-EC-TSLP* algorithm is compared with latest link prediction approaches.

As we mentioned in manuscripts, in the suggested method a genome as well as a group of *LAs* is dedicated to every cell of the input network. The genome corresponds to the predicted connections of the equivalent cell. This is a distributed algorithm in which every genome is locally developed via local information.

In other words, the basis for the proposed algorithm in this article is to weight social network test links according by the selected actions by neighboring nodes. Since the nodes in social networks are dynamic, and also social networks are branching, changes take place in social networks in different time periods.

Neighboring nodes may appear, disappear, or even go unchanged in different time periods. A realization of the network's evolutionary patterns will help us understand the evolution of the network. In this article, we use *ICLA-EC* to analyze network evolution through neighborhood dynamicity. Neighborhood dynamicity in this article is analyzed by using *ICLA-EC*. The dynamicity of neighborhoods will consider temporary neighborhood changes.

The results gained for the proposed algorithm, as well as their comparison with other methods for *Co-authorship, Email-enron, College-MSG* data sets using *AUC* criterion are reported in tables 3, 4 and 5. Please note that these tables give the best outcomes in bold. The results reported here show that the *ICLA-EC-TSLP* algorithm has the ability to achieve the highest *AUC* criterion value for various data sets compared to local similarity criteria such as *CN, AA, PA,* and *JC*. The reason for this superiority is that temporal information is actually an important aspect to consider for link prediction. According to Table 6, the suggested *ICLA-EC-TSLP* can reach *AUC* (0.8010) measure which far surpasses *LP* and *Katz* algorithm. To predict future connections, *LP* and *Katz* algorithms utilize the path length between two nodes, while *ICLA-EC-TSLP* considers temporary neighborhood changes. Our method also surpasses *CMA-ES* algorithm since it employs time series information, while *CMA-ES* just attempts to merge

similarity metrics via defining a weight for every similarity and since the proposed algorithm is much superior to similarity-based methods, it is not surprising that it outdoes *CMA-ES* as well.

Also, it can be seen that *ICLA-EC-TSLP* achieves better *AUC* than the *IP* algorithm. Due to the fact that both algorithms use time series graph information, the superior results produced by the proposed algorithm are due to the fact that *CLA-LP* estimates weight of test links using neighborhood changes that exist in time series graphs, whereas the IP algorithm estimates the future value of any feature in network using a prediction model. The *CLA-LP* algorithm has achieved better results than *MI-LP*, and obviously the reason for that is selecting different neighborhood while moving along different time series, whereas *MI-LP* only uses the mutual node information to predict links. The *ICLA-EC-TSLP* is better than *LA-TSLP* since every vertex utilizes a combination of local information of equivalent vertex and neighboring vertices, while LA-TSLP just employs local similarity based methods to predict future connections. The *ICLA-EC-TSLP* is better than *FLP-DLA* since every vertex utilizes a combination of local information in time series graphs, so the appropriate results spread over the whole population, since neighborhood overlap while *FLP-DLA* just employs strength of test links to predict future connections. *CALA* is a little better than our approach because it considered different similarity metrics and different times with different coefficients.

Table 6. Comparison of AUC of the $ICLA - EC\_TLSP$ with recent Methods on Different Datasets

Method/Dataset	Astro-ph	Hep-ph	Hep-th	Enron Email	College-Msg	Average
LP	0.6322	0.6112	0.6370	0.6554	0.5334	0.6138
Katz	0.6602	0.6527	0.6478	0.6827	0.5529	0.6392
IP	0.6634	0.6478	0.6894	0.6733	0.5665	0.6481
MI-LP	0.7985	0.8034	0.7634	0.7078	0.6994	0.7545
CMA-ES	0.6476	0.6311	0.6581	0.6702	0.5597	0.6333
ARIMA-LP	0.8226	0.8135	0.7716	0.7110	0.7389	0.7715
LA-TSLP	0.7325	0.7947	0.7553	0.7022	0.6945	0.7358
DLA-FLP	0.8297	0.8201	0.7795	0.7212	0.7395	0.7780
CALA	**0.8745**	**0.8527**	**0.7930**	**0.7395**	**0.7795**	**0.8078**
ICLA-EC-TSLP	**0.8650**	**0.8515**	**0.7849**	**0.7212**	**0.7825**	**0.8010**

6. Conclusion

In this paper we presented a novel time series link prediction approach which employs a combination of *ICLA* and *EC* to predict the existence or non-existence of every connection in time period $T + 1$ via the network construction from time periods 1 to T. The traditional static algorithm considers the network within time period T to predict new connections at future time period $T + 1$. However, the proposed algorithm considers time-related information via observation of network evolution over time. In other words, the suggested procedure is an evolutionary algorithm in which every genome corresponds to a part of the solution and the entire population is equivalent to complete solution. In the suggested method a genome as well as a group of *LAs* is dedicated to every cell of the input network. The genome corresponds to the predicted connections of the equivalent cell. This is a distributed algorithm in which every genome, regardless of any previous information, is locally developed via local information. We devised several experiment to analyze the performance of *ICLA-EC-TSLP* compared with other link prediction algorithms. According to the results of the experiments, *ICLA-EC-TSLP* surpasses earlier approaches. In other words, the mixtures of

interactions among vertices as well as the dynamics of the vicinities lead to an improvement in prediction process.

7. 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. Amaral LAN, Scala A, Barthelemy M, Stanley HE (2000) Classes of small-world networks. Proceedings of the national academy of sciences 97 (21):11149-11152
3. Al Hasan M, Zaki MJ (2011) A survey of link prediction in social networks. In: Social network data analytics. Springer, pp 243-275
4. 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.
5. Lichtenwalter RN, Lussier JT, Chawla NV New perspectives and methods in link prediction. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, 2010. ACM, pp 243-252
6. Liben-Nowell D, Kleinberg J (2007) The link-prediction problem for social networks. journal of the Association for Information Science and Technology 58 (7):1019-1031
7. Murata T, Moriyasu S (2008) Link prediction based on structural properties of online social networks. New Generation Computing 26 (3):245-257
8. Dunlavy DM, Kolda TG, Acar E (2011) Temporal link prediction using matrix and tensor factorizations. ACM Transactions on Knowledge Discovery from Data (TKDD) 5 (2):10
9. Whitley D (1995) A review of models for simple genetic algorithms and cellular genetic algorithms. Applications of Modern Heuristic Methods:55-67
10. Baluja S (1992) A massively distributed parallel genetic algorithm. CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE,
11. Baluja S Structure and Performance of Fine-Grain Parallelism in Genetic Search. In: ICGA, 1993. pp 155-162
12. Narendra KS, Thathachar MA (2012) Learning automata: an introduction. Courier Corporation,
13. Thathachar MA, Sastry PS (2011) Networks of learning automata: Techniques for online stochastic optimization. Springer Science & Business Media,
14. Beigy H, Meybodi MR (2004) A mathematical framework for cellular learning automata. Advances in Complex Systems 7 (03n04):295-319
15. Rastegar R, Meybodi MR, Hariri A (2006) A new fine-grained evolutionary algorithm based on cellular learning automata. International Journal of Hybrid Intelligent Systems 3 (2):83-98
16. Esnaashari M, Meybodi MR (2015) Irregular cellular learning automata. IEEE transactions on cybernetics 45 (8):1622-1632
17. Jaccard P (1901) Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bull Soc Vaudoise Sci Nat 37:547-579
18. Adamic LA, Adar E (2003) Friends and neighbors on the web. Social networks 25 (3):211-230
19. Newman ME (2001) Clustering and preferential attachment in growing networks. Physical review E 64 (2):025102
20. Zhou T, Lü L, Zhang Y-C (2009) Predicting missing links via local information. The European Physical Journal B 71 (4):623-630 %@ 1434-6028
21. Katz L (1953) A new status index derived from sociometric analysis. Psychometrika 18 (1):39-43 %@ 0033-3123
22. Moradabadi B, Meybodi MR (2017) A novel time series link prediction method: Learning automata approach. Physica A: Statistical Mechanics and its Applications 482:422-432

23. Bliss CA, Frank MR, Danforth CM, Dodds PS (2014) An evolutionary algorithm approach to link prediction in dynamic social networks. Journal of Computational Science 5 (5):750-764. doi:https://doi.org/10.1016/j.jocs.2014.01.003
24. Rossetti G, Guidotti R, Pennacchioli D, Pedreschi D, Giannotti F Interaction prediction in dynamic networks exploiting community discovery. In, 2015. IEEE, pp 553-558 %@ 1450338542
25. Tan F, Xia Y, Zhu B (2014) Link prediction in complex networks: a mutual information perspective. PloS one 9 (9):e107056
26. Huang Z, Lin DK (2009) The time-series link prediction problem with applications in communication surveillance. INFORMS Journal on Computing 21 (2):286-303
27. Moradabadi B, Meybodi MR (2017) Link prediction in fuzzy social networks using distributed learning automata. Applied Intelligence 47 (3):837-849 %@ 0924-0669X
28. Soares PR, Prudêncio RB (2013) Proximity measures for link prediction based on temporal events. Expert Systems with Applications 40 (16):6652-6660
29. Potgieter A, April KA, Cooke RJ, Osunmakinde IO (2009) Temporality in link prediction: Understanding social complexity. Emergence: Complexity & Organization (E: CO) 11 (1):69-83
30. Janacek G (2010) Time series analysis forecasting and control. Journal of Time Series Analysis 31 (4):303-303
31. Moradabadi B, Meybodi MR (2016) Link prediction based on temporal similarity metrics using continuous action set learning automata. Physica A: Statistical Mechanics and its Applications 460:361-373
32. Özcan A, Öğüdücü ŞG Multivariate temporal link prediction in evolving social networks. In: Computer and Information Science (ICIS), 2015 IEEE/ACIS 14th International Conference on, 2015. IEEE, pp 185-190
33. Thathachar MA, Sastry PS (2002) Varieties of learning automata: an overview. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 32 (6):711-722
34. Wolfram S (1994) Cellular automata and complexity: collected papers, vol 1. Addison-Wesley Reading, MA,
35. Barabâsi A-L, Jeong H, Néda Z, Ravasz E, Schubert A, Vicsek T (2002) Evolution of the social network of scientific collaborations. Physica A: Statistical mechanics and its applications 311 (3-4):590-614
36. Shetty J, Adibi J (2004) The Enron email dataset database schema and brief statistical report. Information sciences institute technical report, University of Southern California 4 (1):120-128
37. Ahn Y-Y, Han S, Kwak H, Moon S, Jeong H Analysis of topological characteristics of huge online social networking services. In: Proceedings of the 16th international conference on World Wide Web, 2007. ACM, pp 835-844