A Knowledge Based Framework for Link Prediction in Social Networks

Pooya Moradian Zadeh(✉) and Ziad Kobti

School of Computer Science, University of Windsor, Windsor, ON, Canada
{moradiap,kobti}@uwindsor.ca

Abstract. Social networks have a dynamic nature so their structures change over time. In this paper, we propose a new evolutionary method to predict the state of a network in the near future by extracting knowledge from its current structure. This method is based on the fact that social networks consist of communities. Observing current state of a given network, the method calculates the probability of a relationship between each pair of individuals who are not directly connected to each other and estimate the chance of being connected in the next time slot. We have tested and compared the method on one synthetic and one large real dataset with 117 185 083 edges. Results show that our method can predict the next state of a network with a high rate of accuracy.

Keywords: Social networks · Link prediction · Cultural algorithm · Evolutionary algorithm · Knowledge · Community detection

1 Introduction

People use social networks to interact with others. Regardless of the content, these interactions can reveal valuable information about real societies and individuals. This information can be useful to identify the structure and topology of these networks, which makes it possible to track their evolutions and predict the next state. Naturally, these networks are extremely dynamic and their rate of evolution is very high. Consequently, their structure changes frequently. Since these networks reflect real life events, having knowledge about their next state can be applied to various domains such as recommendation systems, decision making, marketing and risk analysis [1–4,6,9]. In the field of social network analysis, this problem is known as Link Prediction, which can be defined as estimating the likelihood of a connection between two disconnected entities in a network in the near future [2,4,6].

The main idea behind this problem is, the future state of a network is not random and has a dependency on the current state. Therefore, the target is to find the level of dependency and the main factors affecting it.

Social networks, as a subset of complex networks have some particular characteristics such as power-law distribution and high value of cluster coefficiency. Having a high level of cluster co-efficiency in the network indicates the tendency

M. Gyssens and G. Simari (Eds.): FoIKS 2016, LNCS 9616, pp. 255–268, 2016.
DOI: 10.1007/978-3-319-30024-5_14

of users to join communities is high. Accordingly, in this paper we propose a knowledge-based evolutionary framework based on these properties to estimate the state of a network in the near future just by having one snapshot of the network.

Our proposed model is defined based on the similarity approach with two main assumptions. The first is that an individual in a network tends to join a community. The second is that, according to the homophily phenomenon in social network, each individual joins a community through their friends. Hence the similarity measurement here is defined as having a common community. For example, if a person in a network has 6 friends and 5 of them are members of a community with 30 people. The probability of a friendship between this person and members of the community in the near future is higher than other cases and it can be estimated approximately.

To estimate this likelihood, a knowledge-based structure which is called belief space has been adapted from the evolutionary cultural algorithm which has been proposed for the community detection problem in [13]. Cultural algorithms are a specific type of evolutionary algorithm that use knowledge to enhance the search process to find near optimal solutions for a problem [10,13]. As shown in Fig. 1, a cultural algorithm consists of Population and Belief spaces. In fact, the population space is a list of probable solutions for the community detection problem and the belief space is a knowledge-based structure which guides the population generation process in each iteration and it is evolved by extracting information from the population space [10,13].

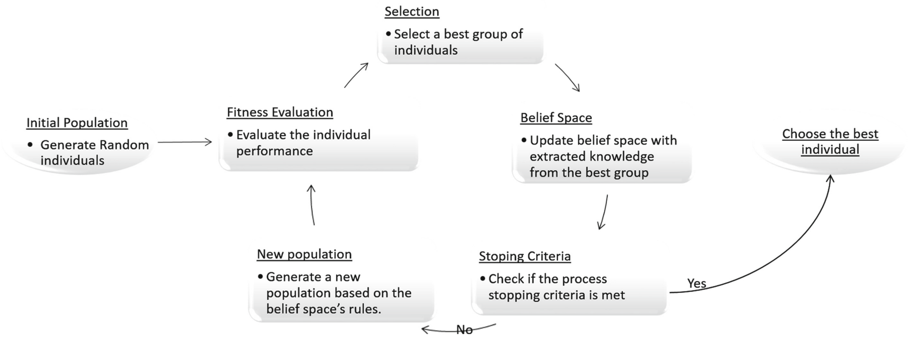

Selection
• Select a best group of individuals
Initial Population
• Generate Random individuals
Fitness Evaluation
• Evaluate the individual performance
Belief Space
• Update belief space with extracted knowledge from the best group
Choose the best individual
New population
• Generate a new population based on the belief space's rules.
Stoping Criteria
• Check if the process stopping criteria is met
Yes
No

Fig. 1. A cultural algorithm process

In this paper, by focusing on the belief space as a great source of knowledge, we propose an algorithm to determine the level of dependency between each pair of users and estimate their tendency to communicate with each other. The main structure of our proposed algorithm is a directed weighted graph which is generated from the belief space data and demonstrates levels of relationships between all neighbor nodes. For predictions, a mathematical formula is proposed to estimate the likelihood of having relationship between each unconnected pair. This formula has been defined based on two main concepts, number of paths between each unconnected pair and length of these paths. Generally having more paths and shorter lengths implies higher chance of connection in the next timeslot. Finally, our algorithm calculates the probability of a relation between pairs of nodes which are not connected together directly and ranks them.

In this research, we present a novel concept of observing the quality of links between pairs of nodes. We also introduce a method to extract information from structure of the network as a similarity index.

The rest of the paper is organized as follows: In the next section, the problem definition and related works will be reviewed. In Sect. 3, we present our model and, after that, the evaluation of the model will be discussed. Conclusions are presented in the last section.

2 Problem Definition and Related Works

If a network maps to a graph, $G(V, E)$, where V is a fixed number of nodes and E represents links between each pair of nodes, an edge is defined as $e = (u, v) \in E$, where $u, v \in V$, at a particular timeslot (t). Predicting a state of the graph at time $t + 1$ by having a snapshot of it at time t, is defined as the Link Prediction Problem in social networks. In other words, given a network G_t at time t, the output of a link prediction algorithm will be a list of edges which are not in G_t and have high probability of appearing in G_{t+1} [2,4,6]. See also Fig. 2.

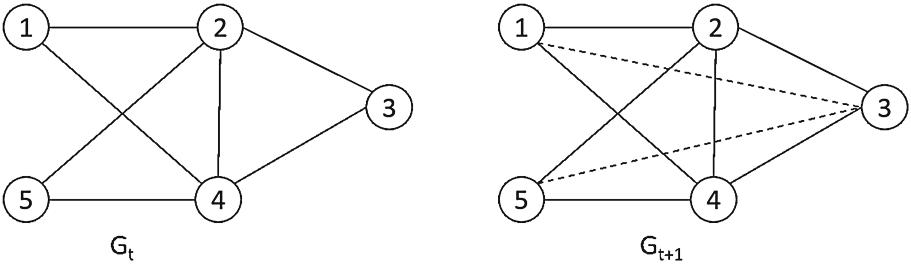


Fig. 2. Predicting the state of a network at time $t + 1$, given a snapshot of it at time t

To solve this problem many studies have been carried out. A similarity-based approach is one of them [4,6]. In fact, having a high number of common features among pair of users increases their chance for making a link in the near future. Therefore, these types of algorithms calculate the level of similarity between each pair of nodes x and node y and assign a score to them. After ranking them, they select the pairs which have higher scores as they have more likelihood to be linked in the near future.

One of the most famous approaches in this field is an unsupervised method which is based on the similarity of the nodes' structure [3,4,6]. To calculate the similarity, many indexes have been proposed, such as the Jaccard similarity co-efficient, Katz, Common Neighbors, Leicht-Holme-Newman, etc. These indexes are mainly based on the number of common neighbors [4,6].

The Common Neighbors index is defined as $C(x, y) = |\mathit{\Gamma}(x) \cap \mathit{\Gamma}(y)|$, where $\mathit{\Gamma}(x)$ and $\mathit{\Gamma}(y)$ are the lists of neighbors of nodes x and y, respectively. This index counts the number of shared neighbors of nodes x and y.

The Jaccard similarity co-efficient is an important index in this field which is defined as $J(x, y) = |\mathit{\Gamma}(x) \cap \mathit{\Gamma}(y)|/|\mathit{\Gamma}(x) \cup \mathit{\Gamma}(y)|$. It measures the number of shared neighbors between two nodes over number of their all unique neighbors.

Leicht-Holme-Newman is also an index which measures the similarity by calculating the number of common neighbors between nodes x and y relative to the product of their degrees: $L(x, y) = |\mathit{\Gamma}(x) \cap \mathit{\Gamma}(y)|/d(x)d(y)$, where $d(x)$ and $d(y)$ are the degrees of nodes x and y, respectively.

The Resource Allocation Index is another index which performs well on real networks. Consider the situation where node x sends some resources to node y through its mutual neighbors. The similarity between x and y is then defined as the amount of resources received by node y: $RA(x, y) = \sum_{z \in \mathit{\Gamma}(x) \cap \mathit{\Gamma}(y)} 1/|\mathit{\Gamma}(z)|$.

However, these indexes are not suitable for all types of networks, their performance is varied based on the structures of different networks [4,6].

In addition, maximum likelihood and probabilistic models approaches which are supervised methods are also used to solve the link prediction problem. However, by increasing the size of the network ($|\text{network}| > 10^4$), these models become impractical because of their time complexity [4,6].

Evolutionary and swarm-based approaches are also used to solve the problem that have been proposed in recent years [1,2,9,11]. In [1], the authors have used the Covariance Matrix Adaption Evolutionary Strategy (CMA-ES) to optimize the prediction accuracy. They suggested a linear model for combining common neighbor's similarity indexes and nodes specific information by assigning a weight to each index. In their model, prior information about the network is not required.

In [2], the authors proposed an algorithm based on ant colony optimization to solve the problem. Random walk strategy has been implemented in their algorithm to select paths. In this algorithm, the probability is assigned to an edge to help an artificial ant select a better edge. In each iteration, the quality of the paths are evaluated to update the probabilities for the next iterations. Finally, the path with higher quality is selected as a link which has more likelihood to appear.

On the other hand, since the future actually is not predictable, to test the accuracy of the algorithm, a network must be randomly divided into two subsets, the training set, E^T, and the probe set, E^P. Here, E^T can be considered as the observed known interactions and E^P as the set of links that must be predicted for testing. In the prediction process, information from E^T must not be used. As a result of this division, $E^T \cup E^P = E$ (the set of the network's edges) and $E^T \cap E^P = \emptyset$.

To evaluate the performance of these algorithms, two main methods are commonly used, the Area Under the Receiver Operating Characteristic Curve (AUC) and Precision [4,6].

For the former, $AUC = (n' + 0.5n'')/n$, where n is the number of independent comparisons and n' denotes the number of times a randomly chosen missing link (a link in E^P) had a higher score than a randomly chosen nonexistent link (a link in $U - E$, where U denotes the universal set containing all possible links, of which there are $|V|(|V| - 1)/2$, with $|V|$ the number of nodes in the network). Furthermore, n'' denotes the number of times that their score is the same [4,6].

For the latter, if the ranked non-observed links are given, Precision is defined as the number of relevant items selected divided by the total number of items selected. In the case that the top-L links from the predicted links are chosen, and L_r denotes the number of these links which are in E^P, then Precision can be defined as L_r/L [4,6].

3 Proposed Evolutionary Model

As we mentioned before, community is the core of our model. Thus, in our model we adapt outputs of the evolutionary cultural algorithm which has been proposed to detect communities on social networks in [13]. While the output of this algorithm is the list of communities, the focus of this research is on the belief space. This belief space can be visualized as a probability matrix which estimates the quality of relationships between each pair of nodes in the network which are directly connected together. Using this belief space which is updated by the extracted information from populations in each iteration, the cultural algorithm limits the search space and enhances the individual evolutions. In our model, we propose using this knowledge repository as a source of information. As shown in Fig. 3, the belief space will map to a directed weighted graph. The weights indicate the level of dependency between each connected pair of nodes. After that, we propose a method to estimate the likelihood of relationships between two unlinked nodes of the graph. Ranking them will be the last process of this model.

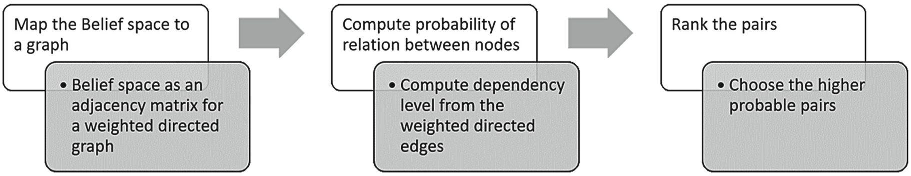

Map the Belief space to a graph
• Belief space as an adjacency matrix for a weighted directed graph
Compute probability of relation between nodes
• Compute dependency level from the weighted directed edges
Rank the pairs
• Choose the higher probable pairs

Fig. 3. Components of the proposed model

3.1 Making the Weighted Graph

First we briefly describe the mentioned cultural algorithm [13]. In this algorithm, an individual is represented as a probable solution based on a particular locus-based adjacency method [8] stored in an array structure. The length of this array is equal to number of nodes in the graph. Each cell of this array is addressed from 1 to n (length of the array) which determines a node in the graph with the same number. E.g., cell #10 corresponds to node #10. For each cell #i, the algorithm will choose an address of a node from the list of neighbors of node #i.

For example, as shown in Fig. 4, if a network has 7 nodes, one sample individual can be defined as an array of nodes, shown in Fig. 5, and illustrated in Fig. 6, which shows two communities in this graph (nodes #1, 5, 6, and 7 in one community and nodes #2, 3, and 4 in another).

As mentioned before and presented in Fig. 1, in each iteration, specific number of individuals are generated by the algorithm (to make a population)

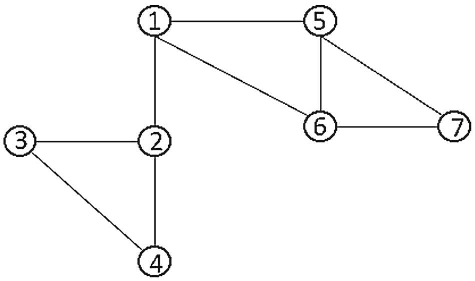
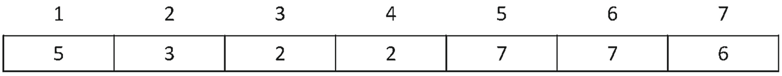


Fig. 4. A sample network



1	2	3	4	5	6	7
5	3	2	2	7	7	6

Fig. 5. A random individual

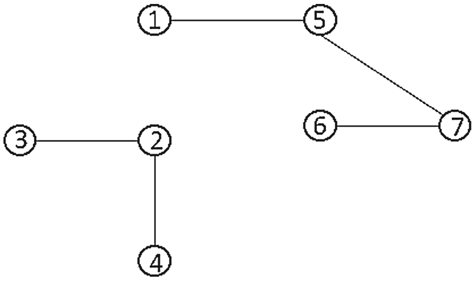


Fig. 6. Illustration of the individual in Fig. 5 which clearly shows two separate communities (1,5,6,7) and (2,3,4)

according to the rules which are set in the belief space. The quality of these individuals is evaluated based on a fitness function. As a result, these individuals can be compared with each other. After sorting them, a group of them that have better fitness values are selected to enter the belief space and if they meet some conditions they can update the belief space.

To update the belief space, each cell of these individuals adds its value to the n by n belief space matrix, where n is the number of nodes, and the algorithm will calculate the relative frequency of it and store it in the matrix as shown in Fig. 7. With this method, the belief space can be considered as an alternative adjacency matrix for the graph, because it is a weighted sub-graph of the main network that shows the level of dependency between nodes according to the community index.

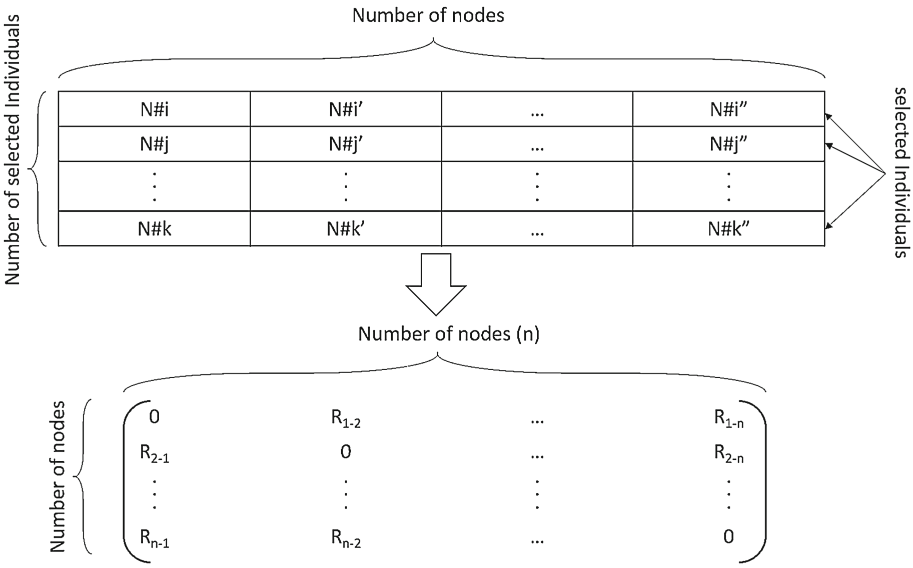
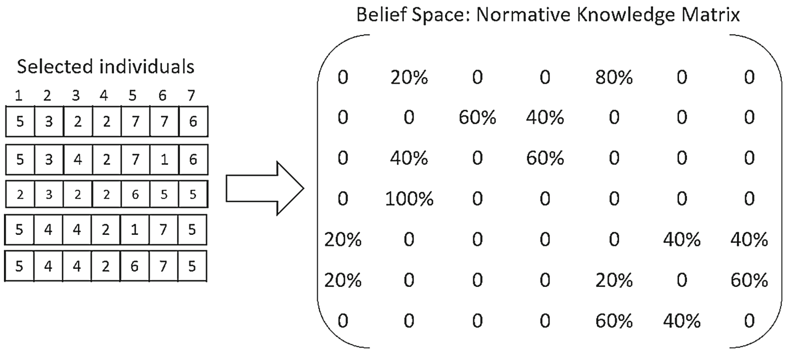


Fig. 7. The structure of the belief space

The belief space plays a key role by setting some rules for generating new generations of individuals. This space collects and saves normative knowledge of the best group of individuals. The assumption is that best individuals are close to an optimal solution, thus the final solution can be generated by combining components of them. In fact, the belief space defines a new state space for the network by storing best individuals. In the subsequent iterations, new generations of individuals are produced mostly based on this state space.

Our main assumption here is, if the number of iterations approaches infinity, the belief space matrix can accurately represent some information about the level of dependency between the connected nodes. Consequently, these relative frequencies can be used as the probability of a relation in the next timeslot



Selected individuals

1	2	3	4	5	6	7
5	3	2	2	7	7	6
5	3	4	2	7	1	6
2	3	2	2	6	5	5
5	4	4	2	1	7	5
5	4	4	2	6	7	5

Belief Space: Normative Knowledge Matrix

$$\begin{pmatrix} 0 & 20\% & 0 & 0 & 80\% & 0 & 0 \\ 0 & 0 & 60\% & 40\% & 0 & 0 & 0 \\ 0 & 40\% & 0 & 60\% & 0 & 0 & 0 \\ 0 & 100\% & 0 & 0 & 0 & 0 & 0 \\ 20\% & 0 & 0 & 0 & 0 & 40\% & 40\% \\ 20\% & 0 & 0 & 0 & 20\% & 0 & 60\% \\ 0 & 0 & 0 & 0 & 60\% & 40\% & 0 \end{pmatrix}$$

Fig. 8. Belief space formed by 5 selected individuals

based on the community function. By processing a snap shot of an undirected and unweighted network, a weighted directed graph is made which reveals hidden information about the quality of relations in the network.

Figure 8 shows an example for updating the belief space. Five individuals have been selected to update the belief space of the same network shown in Fig. 4. If the matrix had been empty before, then it is populated by the relative frequency of nodes and their neighbors. For example, node #5 was linked to node #1, 20 % of times (once out of 5 times). If we illustrate this belief space, as shown in Fig. 9, a directed weighted graph will be the result.

3.2 Computing the Probabilities

To compute the probabilities of relations of a pair of disconnected nodes in this weighted graph, two criteria have been considered to propose a formula. The first is the number of paths between each pairs of disconnected nodes. The second

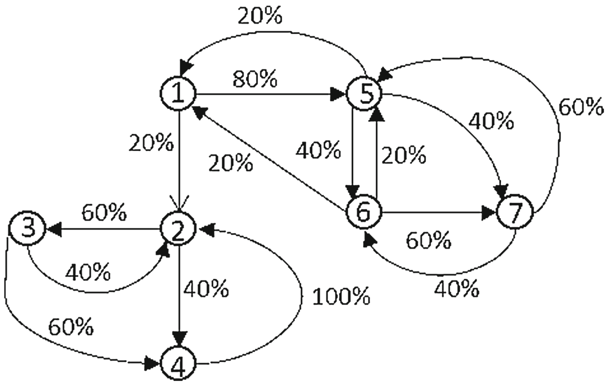


Fig. 9. Illustration of the belief space in Fig. 8

is the length of these paths. To reduce the complexity, we assume the length of the paths is always 1, which means that the probability is computed for those pairs of disconnected nodes that have only one node between themselves. Let $G(V, E, W)$ denote the input weighted graph, where V is a set of nodes and E a set of edges between each pair of nodes (hence, each edge e is of the form (i, j), with $i, j \in V$. Furthermore W is a set of weights of edges, with $0 \leq W(i, j) \leq 1$ for all edges (i, j). For each pair of disconnected nodes (i, k), where $i, k \in V$ and $(i, k) \notin E$, if there is a node j with $j \in V$ and $(i, j), (j, k) \in E$, the estimated weight between i and k is computed as follows:

$$\forall j \in V \rightarrow (i, j), (j, k) \in E, (i, k) \notin E, \\ W'(i, j, k) = \max(W(i, j), W(j, i)) \times \max(W(j, k), W(k, j)). \quad (1)$$

If there were a link between two nodes i and k in the absence of node j, then $W'(i, j, k)$ can be interpreted as the estimated weight of that link.

For each similar path this weight must be computed accordingly, and, finally, the probability of a relation between nodes i and k is computed as follows:

$$P(i, k) = 1 - \frac{1}{2(n + \sum_1^n W'(i, j, k))}, \quad (2)$$

where n is the number of paths between i and k.

For example, in Fig. 9, a direct link does not exist between node#1 and #7 but there are 2 paths of length 1 between them. Therefore, $n = 2$, and the nodes #5 and #6 represent j. We have $W'(1, 5, 7) = 0.8 \times 0.6 = 0.48$ and $W'(1, 6, 7) = 0.2 \times 0.6 = 0.12$, and $P(1, 7) = 1 - (1/(2 \times (2 + 0.6)) = 0.6153$.

3.3 Ranking the Probabilities

After calculating all the probabilities, the predicted pairs must be ranked based on their probabilities. Finally, the top-L of them will be selected as the final predicted edges. This process is shown in the following algorithm:

```
Algorithm CA-LP (G,A,B,L)

Input:
G: an undirected and unweighted graph, G(V,E)
A: adjacency matrix of G
B: Belief Space matrix
L: desired number of top predicted links

Output:
O: n*n matrix of L probabilities where
   O(i,j)=P(i,j), ( i,j are members of V)
```

```
Main:
1: Map Belief space to a weighted directed Graph
2: Compute
P(i,k) by extracting weights from B according to
   (1) and (2), for all pairs where A(i,k)=0 and A(i,j), A(j,k)=1
3: Store probabilities in a array
4: Sort the array
5: Choose the top-L and store in O where O(i,k)=P(i,k)
```

4 Evaluation

To evaluate the performance of the proposed algorithm, we have used one synthetic network and one real large social network dataset. For the synthetic network, 10 graphs were generated randomly based on Newman's method in [7]. Each of these graphs has 128 nodes with degree 16, therefore the graph has 1024 edges. It consists of 4 same-sized communities where each community has 32 members. Each of these members have Z_{in} links to other members who are inside its own community and Z_{out} links to members from other communities ($Z_{in} + Z_{out} = 16$). The range of Z_{out} in these 10 graphs were set from 3 to 5.

As shown in Table 1, we selected 90 % of the graph as E^T and the rest as E^P to evaluate the performance. The belief space which was imported to the algorithm was obtained from the result of running the community detection algorithm proposed in [13]. We tested the effectiveness of the algorithm according to both AUC and Precision methods. The results are illustrated in Table 2 and Fig. 10. Tests were implemented 100 times independently on the top-100 instances. We also compared the results of AUC with three other similarity metrics, Common Neighbors (CN), Jaccard (JC) and Leicht-Holme-Newman (LH).

The results clearly show that the proposed algorithm has better performance in comparison with other metrics on synthetic networks. Another interesting observation is that, by increasing the complexity of the network ($Z_{out} > 4$) the performance of the algorithm reduced significantly. We believe the cause to be the increasing rate of errors in the community detection algorithm when Z_{out} becomes larger.

In addition to Precision, we also compared the top-102 predicted links calculated by the algorithm with the probe set, E^P (|predicted links in $E^P|/|E^P|$). As a result, in average 78.28 % of the predicted links were among the probe set, which means that the algorithm could predict the correct links by an accuracy of more than 75 %.

Table 1. Description of the synthetic network

#Nodes	#Edges	E^T	E^P	U
128	1024	922	102	8128

Table 2. Comparision between different methods

Z_{out}	AUC				Precision
	CA-LP	CN	JC	LH	CA-LP
3	0.901	0.756	0.696	0.899	0.57
3	0.934	0.780	0.754	0.796	0.59
3	0.930	0.893	0.890	0.943	0.63
4	0.963	0.772	0.771	0.957	0.56
4	0.993	0.723	0.623	0.803	0.78
4	0.979	0.801	0.692	0.967	0.70
4	0.955	0.882	0.802	0.940	0.73
5	0.912	0.902	0.800	0.912	0.67
5	0.856	0.834	0.870	0.884	0.65
5	0.895	0.722	0.704	0.809	0.65

We also tested the performance of our proposed algorithm on a big real dataset, Orkut, with 117 185 083 edges [12]. The dataset obtained from the Stanford Large Network Dataset repository [5] is a benchmark dataset used by most researchers in social network analysis. Another reason for selecting this dataset is that it is a network with ground-truth communities which make us possible to validate our results. Information about this dataset is represented in Table 3. The procedure for running the experiment is similar to the procedure described before in experimental setup for synthetic networks. The network was divided

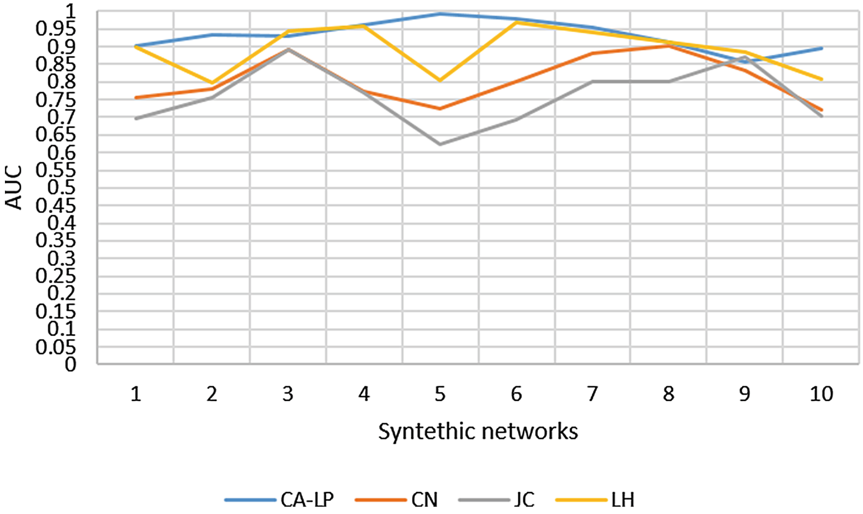


Fig. 10. Comparision of the algorithms based on AUC

Table 3. Orkut dataset specification

#Nodes	#Edges	Cluster coefficiency	E^T	E^P	U
3072441	117185083	0.1666	105466575	11718508	4719945313020

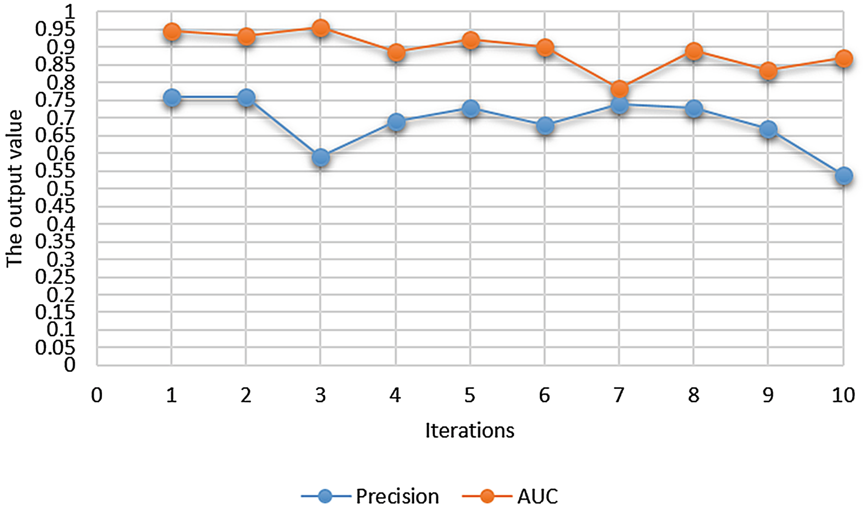


Fig. 11. The results from the orkut dataset

into two sets, the training set (90 %) and the probe set (10 %). After 10 iterations of independent experiments, the AUC and the precision were calculated. As shown in Fig. 11, the algorithm could estimate the correct links with over 68 % success based on the precision method. Regarding the size of the network, we believe that it is an acceptable rate for prediction.

5 Conclusion and Future Work

In this paper, we proposed a knowledge-based model to predict the state of a network in the near future. The key part of this model is the belief space which is a probability matrix that shows the level of dependency between linked nodes. Assuming it as an adjacency matrix, a weighted directed graph can be made. Consequently, the probability of relation between two disconnected nodes will be computed based on this graph.

Estimating the quality of links between a pair of nodes in the network is the first contribution of the algorithm. The second one is defining the concept of community as a similarity index. Finally, the third one is using the cultural algorithm as a knowledge-based evolutionary algorithm to predict the near future.

We evaluated the performance of our algorithm on one synthetic and one large real dataset and compared it with three other metrics. Regarding the results, the algorithm can predict the state of a network with a high accuracy. According to this issue that the objective of evolutionary algorithms is to find near optimal solutions, we believe that by increasing the number of iterations, the quality of prediction will improve. Meanwhile, since the size of the belief space is fixed to the number of nodes, the complexity of the algorithm will not change based on the number of iterations or the number of edges.

In the future, we would like to observe the performance of the algorithm in different type of social networks and extend our work to multiple networks. In addition, currently we have tested the algorithm using the common standard procedure of dividing the training and probe set in the ratio of 90 % and 10 %, in the future we would like to test the performance on different ratios to find the optimal training size.

Acknowledgments. This work is partially supported by a Cross-Border Institute (CBI) Research Grant.

References

1. Bliss, C.A., Frank, M.R., Danforth, C.M., Dodds, P.S.: An evolutionary algorithm approach to link prediction in dynamic social networks. J. Comput. Sci. **5**(5), 750–764 (2014)
2. Chen, B., Chen, L.: A link prediction algorithm based on ant colony optimization. Appl. Intell. **41**(3), 694–708 (2014)
3. Fire, M., Tenenboim, L., Lesser, O., Puzis, R., Rokach, L., Elovici, Y.: Link prediction in social networks using computationally efficient topological features. In: 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), PASSAT/SocialCom 2011, Privacy, Security, Risk and Trust (PASSAT), Boston, MA, USA, pp. 73–80, 9–11 October 2011
4. Hasan, M.A., Zaki, M.J.: A survey of link prediction in social networks. In: Aggarwal, C.C. (ed.) Social Network Data Analytics, pp. 243–275. Springer, USA (2011)
5. Leskovic, J., Krevl, A.: SNAP datasets. In: SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data
6. Lü, L., Zhou, T.: Link prediction in complex networks: a survey. Physica A **390**(6), 1150–1170 (2011)
7. Newman, M.: Detecting community structure in networks. Eur. Phys. J. B **38**(2), 321–330 (2004)
8. Park, Y., Song, M.: A genetic algorithm for clustering problems. In: Koza, J.R., Banzhaf, W., Chellapilla, K., Deb, K., Dorigo, M., Fogel, D.B., Garzon, M.H., Goldberg, D.E., Iba, H., Riolo, R. (eds.) Proceedings of the Third Annual Conference on Genetic Programming, pp. 568–575. Morgan Kaufmann, University of Wisconsin, Madison, Wisconsin, 22–25 July 1998
9. Qiu, B., He, Q., Yen, J.: Evolution of node behavior in link prediction. In: Burgard, W., Roth, D. (eds.) Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, 7–11 August 2011
10. Reynolds, R.G.: An introduction to cultural algorithms. In: Sebald, A.V., Fogel, L.J. (eds.) Proceedings of the Third Annual Conference Evolutionary Programming, pp. 131–139. World Scientific Press, San Diego, CA, 24–26 February 1994
11. Sherkat, E., Rahgozar, M., Asadpour, M.: Structural link prediction based on ant colony approach in social networks. Physica A **419**, 80–94 (2015)

12. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. Knowl. Inf. Syst. **42**(1), 181–213 (2015)
13. Zadeh, P.M., Kobti, Z.: A multi-population cultural algorithm for community detection in social networks. Procedia Comput. Sci. **52**, 342–349 (2015). Shakshuki, E.M. (ed.) Proceedings of the 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015), the 5th International Conference on Sustainable Energy Information Technology (SEIT-2015), London, UK, 2–5 June 2015