IJCAT - International Journal of Computing and Technology, Volume 6, Issue 1, January 2019
ISSN (Online) : 2348-6090
www.IJCAT.org
Impact Factor: 0.835

Finding the Least Significant Edges in the Active Subnets of a Social Network by Dragonfly Algorithm

[1] **Sadegh Saeedi** ; *[2] **Mohammad Hussein Yektaei**

[1] Department of Computer Engineering,
Ahvaz Branch, Islamic Azad University, Ahvaz, Iran

*[2] Department of Computer Engineering,
Abadan Branch, Islamic Azad University, Abadan, Iran

Abstract - The present study aimed at finding the least significant edges based on the target set selection. Selecting a target set consists of a subset of influential and effector nodes in an active social subnet to maximize the influence on the entire nodes of a network. The best heuristic algorithm only ensures that all nodes of a network can be influenced with a maximum of 63% of the target set selection. The concepts of vertex cover were used to find the active and influential nodes. The edges between the active or effector nodes were considered as significant edges while the other edges were considered as least significant. The experiment performed on four laboratory data and a real data showed that the proposed algorithm was more optimized.

Keyword - *Social network, least significant edges, Dragonfly Algorithm, target set selection*

1. Introduction

A fundamental issue in social science, such as religious beliefs, technology acceptance, and success of a new product is a process in which new ideas and behaviors spread through a community. These phenomena tend to start on a small scale with a number of early thinkers and then are adopted by more people because such people see their friends, neighbors or colleagues; and finally, new behaviors may suddenly spread through the population. Such processes are studied as the propagation of information [12]. Skilling and Granvitter expressed the basic mathematical models for mechanisms that shape new ideas and behaviors through a community playing a crucial role during the propagation process [9] [16].

A social network is the base profile of a graph model with nodes representing the agents and edges displaying the interactions between them. [5, 17, 19]. In the 1980s and 1990s, researchers preferred simple networks that connect in regular geometric ways [4] [6]. However, this regular network has some limitations on research attempting to capture the essential aspects of complex social networks. Another simple network topology is a random network in which an individual can be accidentally connected to any other person in the world [8]. However, the previous studies indicated that many real-world networks are systematically different from the topology predicted by random network theory [15]. Wats and Strogasts considered the complex network in the real world passing through a regular network to a random network to reduce the inherent complexity of the real networks [18]. the target set selection problem, which involves discovering a small subset of influential players in a given social network, to perform a certain task of information diffusion [14].

2. The related studies

Richardson and Dominguez (2001) conducted a study on the extraction of significant customer networks by providing a social network model l based on the Markov chain [7]. Camp et al. (2007) studied the discrete optimization based on three models of cascade, independent cascade, weighted cascade, and linear threshold for the first time and stated as the following formula [11, 10]. Leskovec et al. (2007) developed a strategy to find the effective propagations for finding active nodes by greedy algorithms. The speed of this algorithm was 700 compared to the Camp algorithm [13]. Narahari and Narayanam (2011) presented the similar nodes based on the maximum propagation using game theory to optimize the algorithm of Camp and Leskovec. Wang et al. (2014) presented an optimal global method for solving the target set selection by the genetic algorithm.

2.1.Genetic algorithm

Chang Wang et al. concluded that finding the influential and significant nodes using the genetic algorithm had convergent solutions and was solvable [2].

Algorithm 1: John Holland's Genetic Algorithm

```
t ←0
Initialization;
Evaluation;
While not terminal condition do
t ←t +1  ;
Selection;
Crossover;
Mutation;
Evaluation;
End While
```

2.2.Greedy algorithm

Camp et al. proved that the entire nodes of a network can be covered by using the Greedy Algorithm and 63% of nodes Eq. (1) and this is a criterion for finding the significant nodes [10].

$$\left(1-\frac{1}{e}\right) \cong \%63 \qquad (1)$$

In which value e is obtained from Eq. (**2**)

$$e = \sum_{r=1}^{\infty} \frac{1}{r!} \qquad (2)$$

Algorithm 2 : The Greedy algorithm of Camp et al. where N is the node set and K is an integer with value ***K***≤N [10]

```
Set A ← 0
for  i= 1 to k  do
    Choose a node $n_i \in N$  Maximaizing $\sigma(A \cup \{n_i\}) - \sigma(A)$ ;
    $A \leftarrow A \cup \{n_i\}$
End for
```

3. The Proposed Algorithm

The main inspiration of the DA algorithm originates from the static and dynamic swarming behaviors of dragonflies in nature. Two essential phases of optimization, exploration and exploitation, are designed by modelling the social interaction of dragonflies in navigating, searching for foods, and avoiding enemies when swarming dynamically or statistically. Dragonfly Algorithm one of the collective intelligence algorithms working perfectly at in dynamic environments, especially the social network, due to having a dynamic nature in the exploration and exploitation of food. [1]

Algorithm 3 : The Dragonfly algorithm [1]

```
Initialize the dragonfly's population $X_i (i = 1, 2, ., n)$
  Initialize step vectors $\Delta X_i$   $(i = 1, 2... n)$
  While the end condition is not satisfied
      Calculate the objective values of all dragonflies
      Update the food source and enemy
      Update w, s, a, c, f, and e
      Calculate S, A, C, F, and E
      Update neighboring radius
      If a dragonfly has at least one neighboring dragonfly
          Update velocity vector
          Update position vector using
      Else
          Update position vector using
      End if
      Check and correct the new positions based on the boundaries of variables
  End while
End .
```

3.1. Problem statement

Consider the directed network shown in Figure (1), where the black nodes are active and the white nodes are inactive. The activation state of the network is described by an activation vector, **a**.

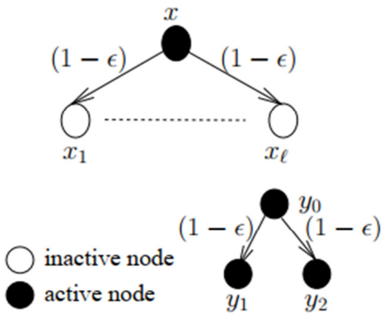

Fig 1: A network with active (black) and inactive (white) nodes. Edge weights represent the probability of an active node activating its neighbors; ϙ ∈ (0, 1) [3]

Assume a simple probabilistic information-propagation model such that every node **V** that becomes active activates a neighbor **X** via a directed link $(v \rightarrow x)$ this activation succeeds with probability equal to the weight of the directed link (v → x). Given a budget k, the propagation started from active Nodes, it would have caused an activation state similar to the one described by **a.**

We call these effector Nodes and the corresponding optimization problem the k-Effectors problem [3]. First we try find a set of k active nodes, and then achieve significant edges between them in the following we can reach the least edges. In **Eq. (3)** The target set selection in social network was minimized after finding the approximation minimum effector node to cover the entire network Nodes, in other words, it is approximation to achieve the optimal minimum target set selection by minimizing approximation of **X**. this can be achieved by solving the vertex cover problem.

$$Min \quad \boldsymbol{C(x)} = \sum_{\boldsymbol{v\in V}} |\boldsymbol{\alpha(v)} - \boldsymbol{\alpha(v.x)}| \qquad (3)$$

Definition 1. Vertex cover in graph G (V. E) of the active subset of $A*$ was selected in such a way that σ as the total members of in Figure (2) covered all graph nodes, i.e. σ (A*) = V where A* is minimum.

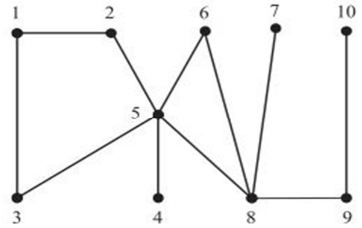

Fig 2: A view of social network with Vertex cover in a connected graph [2]

Example 1. $\sigma(A*)$ = {2,5,8,9} is vertex cover and Significant nodes

In Figure (2) we can have Example.1 The edge between the nodes $\sigma(A*)$ were considered as significant edges and the others as the least significant edges.

Definition 2. Significant nodes are a set of active nodes and effectors nodes which are obtained from the vertex cover

Definition 3. The significant edges are edges which between the collection of vertex cover

Example 1. σ= {(2,5), (5,8), (8,9)}

Definition 4. Target set selection an Approximate selection of an active subnet to influence diffusion the entire network, which should have a minimum node, target set selection is a set of significant nodes.

3.3 Implementation

Its implementation was carried out in three general steps. In the first step, the propagation model was mapped in a social network with a degree centrality approach to a integer linear program Eq.(5). In the second step, the optimized was solved by the metaheuristic algorithm of dragonfly to find the significant nodes. The edge between significant nodes were considered as target set and significant edges. In the third step, the least significant edges were separated from the significant edges based on the data set hypotheses.

$$\begin{array}{ll} \boldsymbol{Min} \ \sum_{\boldsymbol{v\in V}}^{\boldsymbol{n}} \boldsymbol{C(v)X(v)} & \text{minimize the total cost} \\ \forall\{u.v\} \in E \,.\, x_u + x_v \geq 1 & \text{cover every edge of the graph} \\ \forall u \in V \colon x_v \in \{0.1\} & \text{every vertex is either in the vertex cover or not} \end{array} \qquad (5)$$

For this purpose, firstly an initial vector of total vertices was considered as the propagate vertices affecting the entire graph. Such vertices were obtained from the problem hypotheses or the total vertices of the entire graph and were then minimized and optimized based on the behavior of dragonfly algorithm.

3.2. Selecting the optimal solutions

The restrictions of the search spaces under different data sets are summarized in Table (1).

Table 1: Search space limits based on the initial solution set

Dataset	|*N*|	|*N’*|	***r***		
			k = 10	***k = 20***	*k* = 30
Random graph	500	250	9.7656e004	9.5367e007	9.5367e007
Scale-free graph	500	250	9.7656e004	9.5367e007	9.3132e010
Nescience	1589	794	9.7656e004	9.5367e007	9.3132e010

The possibility that one of the optimal target nodes ranks after the top 794 nodes in Nescience data set is quite small [2]. thus the probability of the presence of influential nodes will be much less in the scale higher than 50%. The restrictions of the search spaces under different data sets are summarized in Table (2).

3.3.Simualtion

In the conducted simualtion, the initial population was cosndiered as target set selection in sizes 5 - 10 - 15 - 20 - 25 and 30 with 30 repetitions. In each time, 10,000 random selctions were made where the value 1 was cosndiered for covering the and value 0 was cosndiered for not covering the other vertices.

3.4.The simulator hardware and software specifications

The simualtion was conducetd on a 2GHz Dual-Core Pentium PC and 3GB of original memory with the Windows 7 operating system by using MATLAB software, version 15.

3.5.Shared dataset

The shared data of Table(2) included two laboratory networks and a real network.

Table 2: Summary of the shared data sets in the experiments [2]

Dataset	Nodes	Edges	Average edge	Data type
Random graph 0.005	500	626	2.5	Laboratory
Random graph 0.01	500	1237	4.9	Laboratory
Free scale graph	500	1480	5.9	Laboratory
Nescience graph	1589	2742	3.5	Real

3.5.Evaluation

An evaluation was performed on Table(2). The less number of influential nodes results in less number of active nodes or target set selection. Consequently, the number of the least significnat edges decerases and therefore the number of the least significnat edges will be more optimal in the algorithm Figure(2) The criterion for evaluating the proposed algorithm based on the algorithm of Camp and colleagues was 63% according to equation 1. The proposed algorithm was compared to two greedy algorithms optimzied by the algorithm of Camp and colleagues and the genetic algorithm was conducted on three random graphs and a real network.

- Random graph with 0.005 probability , 500 nodes, and 626 edges

In Table(3) and Figure(3) number of active and influential nodes was more in the entire graph and the obtained least significnate edges was low. Thus, the algorithim was weaker and in selecting the initial popualtion, ten dragonflies were simialr by two algorithms of genetic and Leskovik. In selecting the intial popualtion of more than 10 to 30, the number of active and influential nodes found in the entire graph was low and the number of the least significnat edges was higher. Thus, the algorithm acted optimally.

Table 3 : The laboratory random network with 0.005 probability, 500 nodes, and 626 edges

Algorithm	**Initial popualtion / active nodes**								**Average**	
	0	5	10	15	20	25	30	**Average**	**signifcant**	**Least significant**
Leskovec	0	32	68	96	123	147	170	91	90	536
Genetic	0	40	73	102	128	153	175	96	95	531
Dragonfly	0	60	75	72	72	68	68	59	58	567

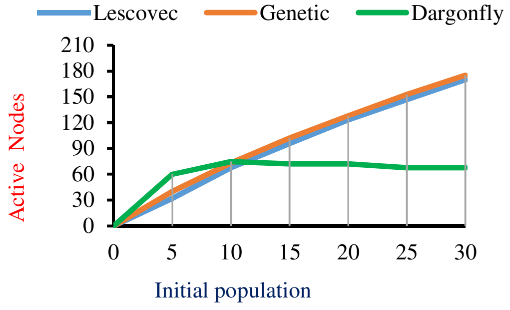

Fig 3: The laboratory random graph with 0.005 probability, 500 nodes, and 626 edges

- Random graph with 0.01 probability , 500 nodes, and 1237 edges

In Table(4) and Figure(4) of the proposed algorithm with intial popualtion of five dragonflies, the number of active and influential nodes in the entire graph was more than Leskovec and genetic algorithms. Thus, its least significnate edges was lower. Thus, the algorithim was weaker but in the intial popualtion of more than 10 to 30, the number of active and influential nodes found in the entire graph was lower and the number of the least significnat edges was higher.

Table 4: The random network with 0.01 probability , 500 nodes, and 1237 edges

Algorithm	Initial popualtion / active nodes								Average	
	0	5	10	15	20	25	30	Average	signifcant	Least significant
Leskovec	0	44	90	135	165	186	195	116	115	1122
Genetic	0	54	104	145	175	197	210	126	125	1112
Dragonfly	0	60	75	72	72	68	68	59	58	1179

, 500 nodes, and 1237 edges

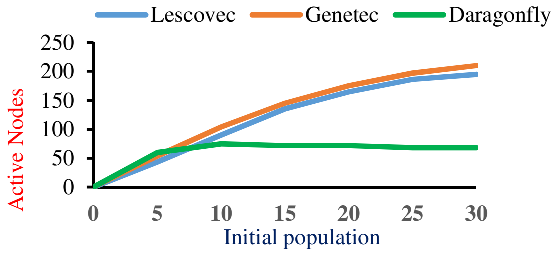

Figure 4: The random graph with 0.01 probability

- Laboratory graph of scale free network with 500 nodes and 1480 edges

In Tabl.5 and Figure.5 of the proposed algorithm with intial popualtion of 5-30 dragonflies, the number of active and influential nodes in the entire graph was more than Leskovec and genetic algorithms while the number of the least significnat edges was higher. Thus, the algorithm was optimal**.**

Table 5 : The laboratory graph free scale netwrok with 500 nodes, and 1480 edges

Algorithm	Initial popualtion / active nodes								Average	
	0	5	10	15	20	25	30	Averag	signifcnat	Least significnat
Leskovec	0	140	210	250	280	300	310	213	212	1268
Genetic	0	145	218	260	290	310	320	220	219	1261
Dragonfly	0	60	75	72	72	68	68	59	58	1422

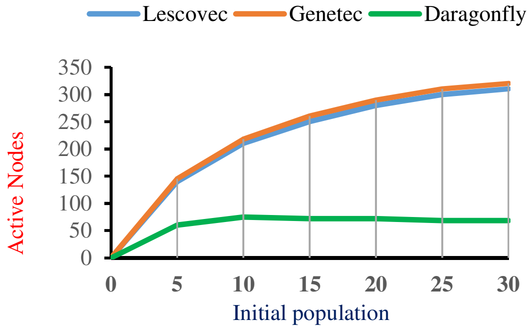

Figure 5 : free scale graph with 500 nodes, and 1480 edges

- Real Nescience graph with 1589 nodes and 2742 edges

In Table(6) and Figure(6) of the proposed algorithm with intial popualtion of 5-10 dragonflies, the number of active and influential nodes in the entire graph was more than Leskovec and genetic algorithms. Thus, the algorithim was wekar but in the popualtion of 15-25, the algorthim was tangent and ahd competition. In then popualtion of 30 dragonflies, te algortihm was better than Leskovec and genetic algorithms. Thus, the algrothim was optimal.

Table 6: Real Nescience graph with 1589 nodes and 2742 edges

Algorithm	Initial popualtion / active nodes								Average	
	0	5	10	15	20	**25**	**30**	Average	signifcnat	Least significnat
Leskovec	0	70	130	174	214	244	270	157	156	2586
Genetic	0	80	141	184	225	255	280	166	165	2577
Dragonfly	0	100	150	180	220	250	260	166	235	2507

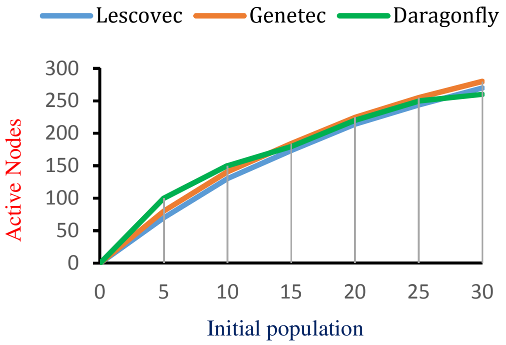

Figure.6. Real Nescience graph

- The comparsion graph for finding the least signifcnat edges

Table(4) showed a comaprison between three algorithms of Leskovec , genetic, and dragonfly on four models of Table(2) in finding the least signifcnat edges.

4. Total Resulting

Figure 6 showed that the proposed algrotihm acted optimally in comaprison to 0.01 and 0.005 random graphs and fere scale graph in tersm of finding the number of least significnat edges. Furtehmore, the proposed algrotihm was shown weaker than Leskovik and genetic algorithms with a small percentage.

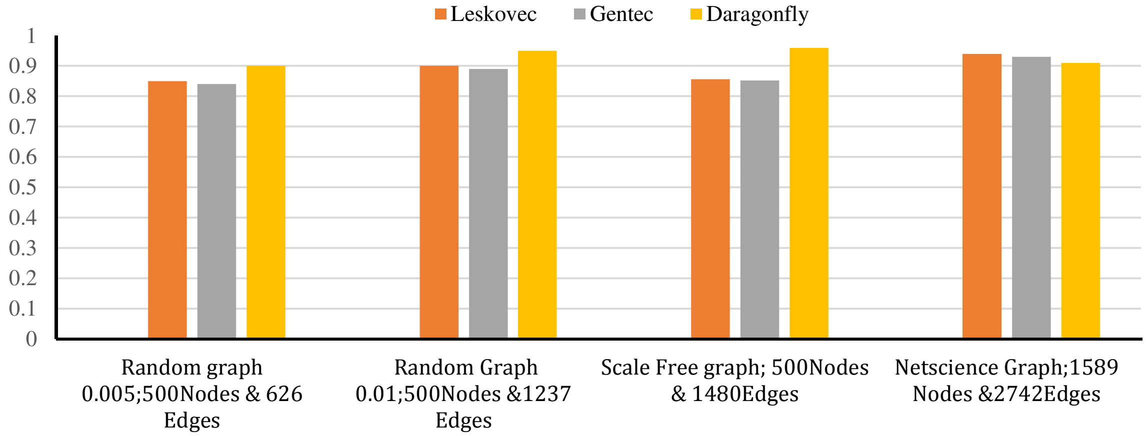

Figure7: Comparing the least significnat edges from table 2

5.Conclusion

Today networks are widespread everywhere Spread of the networks in the world, It forces us to study them, Their study is based on the method of creating nodes and their relationship in the form of a model. The purpose of this article is to find least-significant edges.the least-significant edges will be obtained from significant-edges.Finding the significant-edges is a Np-hard problem.significant-nodes is same effectors node and also significant-edges is k-effectors. Our suggestion is to use vertex cover to find significant- nodes.The proposed algorithm was an daragonfly algorithm.Their fitness alghorithm was obtained using an integer linear programming model. In the evaluation, compared with three labratory networks and a real network, the proposed Dragonfly algorithm obtained stronger than the greedy and genetic algorithm.

References

[1] Mirjalili, S.” Dragon fly algorithm: a new meta-heuristic optimization technique for solving single-objective,discrete,and multi-objective problems”, Neural computation &Application ,27:1053:1073 ,2015.

[2] Cheng Wang, Lili Deng, Gengui Zhou, Meixian Jiang " A global optimization algorithm for target set selection problems”, Information Sciences: An International Journal: Volume 267, May, Elsevier Science Inc 2014

[3] Lapps T.,Terzi E.,” Finding Effectors in social Networks”,ACM,25-28, ,2010.

[4] R. Axelrod, The dissemination of culture: “a model with local convergence and global polarization”, The Journal of Conflict Resolution 41 (2) (1997) 203–226

[5] F. Buccafurri, V.D. Foti, G. Lax, A. Nocera, D. Ursino, “Bridge analysis in a social internetworking scenario”, Information Sciences 224 (2013) 1–18.

[6] R. Cowan, J. Miller, “Technological standards with local externalities and decentralized”

[7] P. Domingo’s, M. Richardson, “Mining the network value of customers”, in: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 502525, 2001, pp. 57–66.

[8] E. Gilbert,” Random graphs “, The Annals of Mathematical Statistics 30 (1959) 1141–1144.

[9] M. Granovetter , “Threshold models of collective behavior”, American Journal of Sociology 83 (6) (1978) 1420–1443.

[10] D. Kempe , J. Kleinberg, E. Tardos, “Maximizing the spread of influence through a social network”, Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 956769, ACM, 2003, pp. 137–146.

[11] D. Kempe, J. Kleinberg, E. Tardos, “Influential nodes in a diffusion model for social networks”, in: L. Caires, G. Italiano, L. Monteiro, C. Palamidessi, M. Yung (Eds.), Automata, Languages and Programming, Lecture Notes in Computer Science, vol. 3580, Springer, Berlin, 2005, pp. 1127–1138.

[12] J. Kleinberg, “Cascading behavior in networks: algorithmic and economic issues “, in: N. Nisan, T. Rough garden, E. Tardos, V.V. Vazirani (Eds.), “Algorithmic Game Theory”, Cambridge University Press, New York, 2007, pp. 613–632.

[13] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, “Cost-effective outbreak detection in networks”, Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 1281239, ACM, 2007, pp. 420–429.

[14] R. Narayanam, Y. Narahari, “A shapley value-based approach to discover influential nodes in social networks”, IEEE Transactions on Automation Science and Engineering 8 (1) (2011) 130–147.

[15] E. Ravasz, A. Barabási, “Hierarchical organization in complex networks”, Physical Review E 67 (2) (2003) 026112.

[16] T. Schelling, “Micro motives and Macro behavior”, Norton, 1978.

[17] S. Strogatz, “Exploring complex networks”, Nature 410 (8) (2001) 268–276.

[18] D. Watts, S. Strogatz, Collective dynamics of’ ’small-world’’ networks, Nature 393 (4) (1998) 440–442.

[19] S.J. Yu, “The dynamic competitive recommendation algorithm in social network services”, Information Sciences 187 (2012) 1–14.

Authors -

First Author: Sadegh Saeedi MSc student of computer engineering Department of Computer Engineering, Ahvaz Branch, Islamic Azad University, Iran.

* **Second Author:** Dr. Mohammad Hossein Yektaei ; Ph.D. Assistant Professor, Department of Computer Engineering, Abadan Branch, Islamic Azad University, Iran.