Bitlis Eren Üniversitesi Fen Bilimleri Dergisi
BİTLİS EREN UNIVERSITY JOURNAL OF SCIENCE
ISSN: 2147-3129/e-ISSN: 2147-3188
VOLUME: 13 NO: 2 PAGE: 408-417 YEAR: 2024
DOI:10.17798/bitlisfen.1407941

Estimating the Expected Influence Capacities of Nodes in Complex Networks under the Susceptible-Infectious-Recovered Model

Aybike ŞİMŞEK[1*]

[1]*National Defence University Turkish Military Academy, Department of Computer Engineering*
Devlet Mah. Kara Harp Okulu Cd. 06420 Ankara / Türkiye,
(ORCID:0000-0002-1033-1597)

Keywords: Complex networks, Susceptible-Infectious-Recovered model, Expected influence, Centrality Measures, Node ranking.

Abstract

In recent years, epidemic modeling in complex networks has found many applications, including modeling of information or gossip spread in online social networks, modeling of malware spread in communication networks, and the most recent model of the COVID-19 pandemic. If the information disseminated is accurate, for example, maximizing its distribution is desirable, whereas if it is a rumor or a virus, its spread should be minimized. In this context, it is very important to identify super-spreaders that maximize or minimize propagation. Lately, studies for detecting super-spreaders have gained momentum. Most of the studies carried out aim to distinguish the influences of nodes under a specific propagation model (such as SIR) using network centrality measures and subsequently, to rank the nodes accordingly. However, in this study, we developed an algorithm that approximates the expected influence of nodes under the popular SIR model. By considering the behavior of the SIR model and only the shortest paths between nodes, the algorithm ranks the nodes according to this approximated value. Our developed algorithm is named the Expected Value Estimation (EVE). The main contribution of this study is that under the SIR model, the effects of nodes can be calculated quickly and realistically, regardless of the structure of the network. We compared the performance of EVE, using different SIR settings on real-world datasets, with that of many current well-known centrality measures. The experimental studies demonstrated that the solution quality (ranking capability) of EVE is superior to that of its competitors.

1. Introduction

Complex networks are highly suitable tools for modeling the real world. They have applications in many different fields such as natural sciences [1], health [2], cyber security [3], economics [4], and social networks [5]–[7]. Moreover, epidemic modeling in complex networks has attracted attention in recent years for its many practical benefits. The spread of a virus outbreak (such as Covid-19) can be estimated and precautions can be taken based on this [8]. By modeling the spread of gossip on the social network, the spread can be prevented [9], [10]. Or, the desired information may reach the maximum number of people [11]. Whether you want to minimize the spread of gossip or maximize the spread of information, in any case, in order to do so, the set having the smallest number of the most influential individuals should be identified [12], [13]. The influences of these individuals under certain epidemic models (such as SIR) should be calculated in order to identify the smallest number of the most influential individuals (i.e., key players). For this, it is necessary to model the propagation by selecting each node individually as the seed. Since propagation models are stochastic models, they must be repeated many times (e.g., about 10.000 iterations) and the average value taken. This operation requires very high

---

[*] Corresponding author: aysimsek@kho.msu.edu.tr

Received: 21.12.2023, Accepted: 20.03.2024

processing power. On the other hand, researchers have noticed a correlation between the influence capacity of the nodes and network centrality measures, which have been used for a long time to determine the importance of nodes in complex networks. The basic expectation here is that as a centrality measure increases, the influence capacity increases, and as the centrality measure decreases, the influence capacity decreases. Since the calculation of centrality measures requires much less processing power than modeling the propagation thousands of times, studies have turned to this area. For this purpose, basic centrality measures such as Degree, Closeness, Betweenness [14], Katz [15], PageRank [16] were used and new centrality measures were developed. However, many of the measures developed only considered the local and global impacts of the nodes [17]–[22] or network communities [18], [19], [23]–[25]. Recently, another approach has been adopted that combines multiple centrality measures to develop new hybrid centrality measures [26]–[34]. However, many of these studies ignore the dynamics of the propagation model. Unlike other studies the dynamics-sensitive (DS) centrality combines network structure and epidemic model dynamics for ranking nodes [35]. The DS considers all possible random walks between two nodes for estimating the infectious probabilities of nodes. This method may work well for small beta values. However, it will overestimate the infection capabilities of nodes for larger beta values.

In this study, we developed an algorithm that ranks nodes according to their influence capacity, taking into account the propagation behavior in the Susceptible-Infectious-Recovered (SIR) model. We named our developed algorithm the Expected Value Estimation (EVE) because it is based on approximating the expected influence of each node. It is worth mentioning here that the EVE algorithm does not calculate the importance of nodes contrary to the centrality measures. Instead, it calculates the approximate expected influence of the nodes under the SIR model and ranks the nodes accordingly.

1.1. Motivation

In the literature, centrality measures are proposed to detect influential nodes in complex networks. However; complex networks show very diverse characteristics, and therefore while a particular centrality measure can well distinguish influential nodes in a given network; it may not be able to on another network. In addition, the influence levels of nodes in the same network may vary under different propagation models. Therefore, trying to determine the influence levels of nodes only based on their centrality will not yield successful results in every network. When determining the influence levels of nodes, it is necessary to consider the dynamics of the given propagation model. Under certain epidemic models (such as SIR), it is necessary to perform heavy Monte-Carlo simulations to distinguish the influence of nodes However, if the dynamics of the SIR propagation model are taken into account, the process can be simplified by ignoring some of the behaviors of this model. Thus, the approximate expected influence of nodes can be calculated and used to rank nodes (similar to a centrality measures). With this motivation, by considering the behavior of the SIR model and only the shortest paths between nodes, the EVE algorithm ranks the nodes according to this approximated value.

2. Preliminaries

Before discussing the details of EVE, it would be useful to give some preliminary information.

Let $G = (V, E)$ be an undirected unweighted graph (network). Here, V is the set of nodes (vertices), and E is the set of edges (links).

2.1. Definition 1 (Susceptible-Infectious-Recovered Model)

The Susceptible-Infectious-Recovered (SIR) model is a well-known model used for population-based epidemic modeling. In recent years, due to their popularity, SIR and SIR variations have been applied to network topologies [36]. In the SIR model, nodes are found in one of three states: Susceptible (S), Infected (I), and Recovered (R). Nodes in state S are nodes that have the potential to become infected. Nodes in state I are nodes that have already been infected. Nodes in state R are recovered nodes. The transition of nodes between states occurs according to certain probabilities. A node in state I continues to infect its neighbors in state S with a certain probability as long as it remains in state I. This probability of infection is known as β. In other words; Nodes in state S are infected with probability β by nodes in state I. Once a node goes to state I, it cannot return to state S again. Nodes in state I goes to state R with a certain probability. This probability of recovery is known as γ. Once a node goes to state R, it cannot return to either state S or state I again.

Initially, all other nodes are in a susceptible state, except for nodes that carry the disease (i.e., those that are infected). Starting from the nodes that are initially infected (called ‘seed nodes’), the disease spreads over the network. After a certain period of time, there are no remaining infected nodes on the network and thus, the model is terminated.

2.2. Definition 2 (Kendall's tau Ranking Correlation Coefficient) [37]

Let (a_i, b_i) and (a_j, b_j) be tuples of joint A and B ranking lists. If $a_i > a_j$ and $b_i > b_j$ or $a_i < a_j$ and $b_i < b_j$, then the tuples are concordant. If $a_i > a_j$ and $b_i < b_j$ or $a_i < a_j$ and $b_i > b_j$, then the tuples are discordant. If $a_i = a_j$ or $b_i = b_j$, then the tuples are neither concordant nor discordant. Finally, tau is defined as in Equation (1).

$$tau = \frac{N_c - N_d}{0.5N(N-1)} \quad (1)$$

Here, N_c is the number of concordant pairs, N_d is the number of discordant pairs, and N is the number of all combinations. Positive tau values indicate a positive correlation, and negative tau values indicate a negative correlation.

2.3. Definition 3 (Ranking Monotonicity) [38]

When you add a figure to your article, please refer to the relevant picture in the text, such as Figure 1. When using shapes, be attentive to use the Shape Description style. Additionally, there should be a 5 nk space between the figure and its caption [6], [7]. Monotony is a metric of how well the centrality measure assigns each node to different rank levels. The ranking monotonicity (RM) will be '1' if all nodes are assigned to a different ranking level. If all nodes are assigned to the same ranking level, the RM will be '0'. Of course, for a centrality measure, the closer it is to RM 1, the better. The RM is calculated as follows:

$$RM(L) = \left(1 - \frac{\sum_{r \in L} n_r(n_r - 1)}{n(n-1)}\right)^2 \quad (2)$$

Here, n is the length of the L-ranking list and n_r is the number of elements assigned to the same r rank.

3. EVE

Generally speaking, in the SIR model, a node affects its neighbor nodes with a probability β. If not its direct neighbor, it is likely to affect its neighbors' neighbors with probability (β × β). If the network is a tree, the probability of a node influencing another l-hop away node can be calculated as β^l since there can be only one path between each pair of nodes. Thus, the expected influence of a node can be calculated using its distance to all other reachable nodes by this node as the sum of β^l values. However, real networks rarely exhibit tree structures. Hence, there can be many different paths of different lengths between any two nodes. It is also costly to use all paths to all other nodes to calculate the expected influence of a node. However, the probability of one node influencing another node decreases exponentially with the distance between them, although in practice, the value of β is much less than 1. The natural consequence of this is $\beta^n \gg \beta^{n+1}$, where $n \in \mathbb{N}^+$. Based on this information, the expected probability of a node influencing another node can only be approximated using the shortest path between these two nodes. This is because the probability of influence calculated for paths other than the shortest path will be much lower. The calculated values can be used to distinguish the influence capacities of the nodes (similar to a centrality measure).

The working principle of EVE is based on expected value calculation. Therefore, it is useful to first look into the details of how a node infects its neighbor nodes in SIR and how this node recovers. This situation is shown for one iteration in Algorithm 1 [39]. The node u in the algorithm was initially selected as the infected node or one infected at any point in time.

Algorithm 1. Infection and Recover States of SIR

```
sn = susceptible neighbors of node u
for each v in sn
    rnd = random number in [0.0,1.0)
    if rnd < $\beta$ then
        mark v as infected
end for
rnd = random number in [0.0,1.0)
if rnd < $\gamma$ then
    mark u as recovered
```

According to Algorithm 1, the node u infects its neighbors with probability β. After the node u infected its neighbors, this node is recovered with probability γ. If γ = 1, the node u has absolutely only one attempt to infect its neighbors since it will not be in the Infected state in the next iteration. If γ = 0.5, roughly, the node u has two attempts to infect its neighbors since it will be in the Infected state in the next iteration with probability 0.5. If we generalize, the node u has at least $1/\gamma$ attempts to infect its neighbors. Since the probability of the node u infecting its neighbors is β, the expected value of infecting a neighbor by node u would be $1/\gamma$ times β; that is, β/γ.

Let us explain the situation in Figure 1, where different topologies are shown. Notice that Figure 1-a, b, and c are trees. Therefore, there is only one path between all nodes.

In Figure 1-a, let the node u initially be selected as a seed (infected). The expected influence value (ev) of the node u becomes ev(u) = $1 + \beta/\gamma$. Here, 1 has been added as node u is already infected. Figure 1-b shows the expected influence value (ev) of the node as:

$$\text{ev(u)} = 1 + {}^{\beta}/_{\gamma}\ (\text{probability of } u \text{ infecting } y).$$

In order to infect the node y, the node u must infect the node x. Next, the node x must infect the node y. The probability of these two events happening together can be obtained by multiplying the probabilities of their respective occurrence. Thus, the expected value of u infecting the node y is $\left({}^{\beta}/_{\gamma} \times {}^{\beta}/_{\gamma}\right)$, i.e., $\left({}^{\beta}/_{\gamma}\right)^2$. Thus, the expected influence value (ev) of the node u becomes ev(u) = $1 + {}^{\beta}/_{\gamma} + \left({}^{\beta}/_{\gamma}\right)^2$.

For Figure 1-c, the expected influence value (ev) of the node u is ev(u) = $1 + 2 \times \left({}^{\beta}/_{\gamma}\right) + 2 \times \left({}^{\beta}/_{\gamma}\right)^2$.

The expected value of a node infecting another node decreases exponentially with the distance between them. If we generalize the ev calculation, we get Equation (3).

$$ev(u) = 1 + nn_1 \times \left({}^{\beta}/_{\gamma}\right) + nn_2 \times \left({}^{\beta}/_{\gamma}\right)^2 + \cdots + nn_h \times \left({}^{\beta}/_{\gamma}\right)^h \qquad (3)$$

Here, nn is the size of the set of node u's neighbors at h-hop distance. The situation is a little different in Figure 1-d. The node y is both a 1-hop and a 2-hop neighbor of the node u. Therefore, the node u can infect the node y directly, as well as through the node x. Thus, the expected value of node u infecting the node y is the sum of these two possibilities, or 1 at most. Ultimately, the expected influence of the node u becomes ev(u) = $1 + \left({}^{\beta}/_{\gamma}\right) + max\left\{1, \left(\left({}^{\beta}/_{\gamma}\right) + \left({}^{\beta}/_{\gamma}\right)^2\right)\right\}$. Let us explain why we use the max function here. For example, if ${}^{\beta}/_{\gamma} = 1$, the expected value of node u infecting the node y would be 2. However, this value can be at most 1, since once a node is infected, it cannot be infected again.

In large and complex networks, there can be many different paths having different lengths from one node to another. It is quite costly to consider all paths. Instead, only the shortest paths can be considered. Thus, as in Figure 1-e, the (x, y) edge is ignored and the approximate ev can be calculated using Equation (3). However, instead of changing the structure of the graph, only neighbors with h-shortest path hop distance can be included when creating nn_hsets. Thus, it is guaranteed that $nn_a \cap nn_b = \emptyset$; here $a \neq b$ and $a, b \in \{1 \ldots h\}$. If we named as spn_h to the sets created by selecting only neighbors with h-shortest path hop distance, we can calculate the measure we call EVE as in Equation (4).

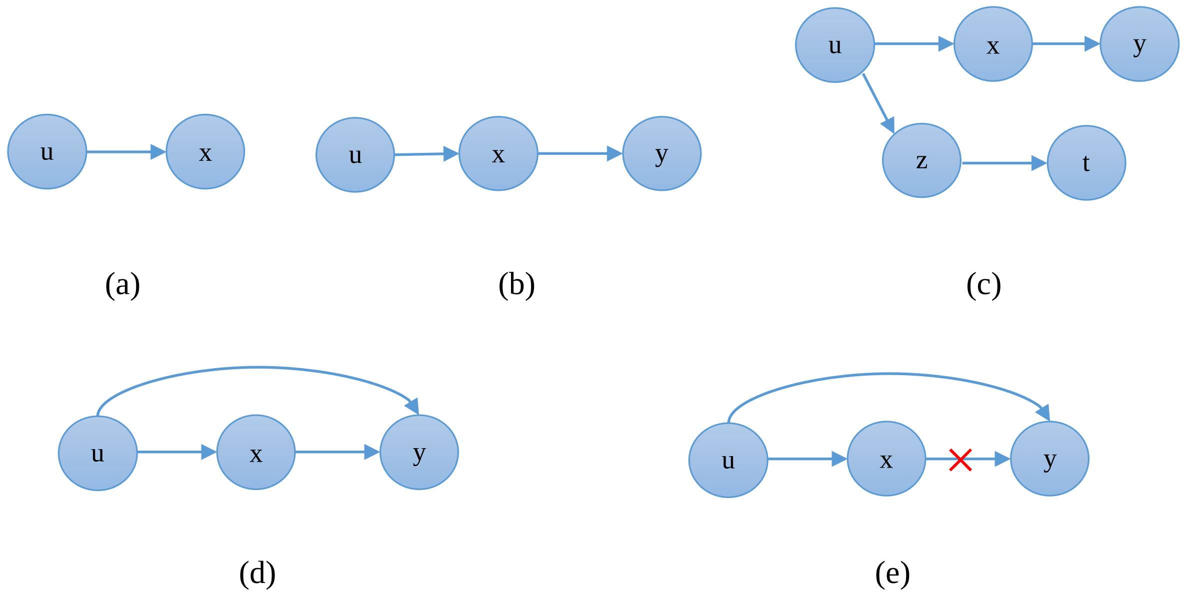
$$EVE(u) = 1 + spn_1 \times \left({}^{\beta}/_{\gamma}\right) + spn_2 \times \left({}^{\beta}/_{\gamma}\right)^2 + \cdots + spn_h \times \left({}^{\beta}/_{\gamma}\right)^h \qquad (4)$$

Equation (4) does not take into account paths other than the shortest paths. In the literature, β is usually taken as very small (e.g., ≤0.1) and **γ** as large (e.g., = 1). The corollary of this is $\left({}^{\boldsymbol{\beta}}/_{\boldsymbol{\gamma}}\right)^{\boldsymbol{l}} \gg \left({}^{\boldsymbol{\beta}}/_{\boldsymbol{\gamma}}\right)^{\boldsymbol{l+1}}$, where $\boldsymbol{l} \in \mathbb{N}^+$. Thus, it can be considered reasonable to ignore paths other than the shortest paths. In practice, EVE can be calculated as in Algorithm 2. The Sort function sorts the dictionary in descending order. The Power function takes two parameters such as x and y and returns the value x^y. As a result, Function EVE returns the list of nodes sorted in descending order according to their EVE values.

(3)

Algorithm 2. EVE

```
FunctionEVE(G: Graph, $\beta, \gamma$)
Begin
L = {} // L is a dictionary as L[node]=EVE
SP = dictionary of all pairs shortest path of G.
// If there is at least one path between two nodes then
// SP[node,node] is a number. // Otherwise, it is ∞.
V = G's set of nodes
for each u in V
   EVE = 0
   for each v in V
      if SP[u,v]≠ ∞ then
         EVE = EVE + Power($\beta$ / $\gamma$, SP[u,v])
      L[node]=EVE
Sort L descending order by value
return key list of L
End
```



(a) (b) (c) (d) (e)

Figure 1. Sample graphs for EVE calculation: (a), (b), (c) every node belongs to only one h-hop neighborhood, (d), (e) node-y belongs to different h-hop neighborhoods.

4. Experiments

To evaluate the performance of EVE, we determined five competitor centrality measures and experimented with different SIR settings over four real-world datasets. First, let us look at the competing centrality measures and datasets.

4.1. Centrality Measures

DC (Degree Centrality) is calculated by dividing the degree of the node by the total number of nodes in the graph minus one [40].

EC (Eigenvector Centrality) is used to determine the importance of a node in the network. The basic logic of EC is that the more adjacent a node is to the important nodes, the more important it is [41].

CC (Closeness Centrality) is a measure of how close a node is to other nodes [42]. The closer the node is to other nodes, the larger the CC.

BC (Betweenness Centrality) is the proportional information on how many of the shortest paths between all pairs are through a node [14].

GC (Gravitational Centrality) is a recent centrality measure inspired by Newton's gravitational formula [30]. Instead of the mass in the original formula, it uses the k-shell values of the nodes and instead of the distance, it uses the length of the shortest path between nodes. Its formula is as follows:

$$GC_i = \frac{ks_i \times ks_j}{\sum_{j \in \mathrm{N}} d(j,i)} \tag{5}$$

Here, $d(\cdot)$ is the length of the shortest path between nodes i and j; N is the set of 3-hop neighbors of node i.

MLD (multi-local dimension) is a state-of-the-art centrality measure proposed by Wen et. al [43]. MLD considers a node as the center and calculates the ratio of this node's neighbors up to the r-hop distance to the number of all nodes for different radius (r) values. It then calculates a centrality measure for the node based on this value. For the details of MLD, the related study can be examined.

4.2. Datasets

We used the following one synthetic and eight real-world networks for the experiments. The properties of the networks are given in Table 1. All the real-world datasets are taken from http://networkrepository.com [44].

Table 1. Network dataset features.

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	$\langle K\rangle$	K_{max}
Barabasi-Albert	1000	9900	19.8	198
Ca-GrQc	5242	14496	5.53	81
Email-Enron	143	623	8	42
Email-Univ	1133	5451	9.62	71
inf-power	4941	6594	2.66	19
inf-USAir97	332	2126	12.80	139
rt_alwefaq	4171	7123	3.41	879
rt_bahrain	4676	8007	3.42	261
rt_damascus	3052	3881	2.54	648

4.3. Performance Comparison of the Measures

We evaluated the performance of EVE and the competitor centrality measures from different angles. First, we looked at the Kendall ranking performances. We then compared their Monotonicity performances. Finally, we looked at how many of the nodes in the top 5% of the ranking lists created by the measures corresponded to the ranking lists created according to the SIR simulations.

We applied SIR model to measure influences of nodes. We set $\gamma = 1$, and we tried different values for β around the epidemic threshold (β_{th}). The epidemic threshold is calculated as in (6) [45].

$$\beta_{th} \approx \frac{\langle k\rangle}{\langle k^2\rangle - \langle k\rangle} \qquad (6)$$

Here ⟨k⟩ denotes the average degree, and ⟨k2⟩ denotes the second-order moment of the degree distribution [45].

In the SIR simulations, we set each node as the only infected node in the network. We ended the simulations when there were no infected nodes left in the network. At the end of each simulation, we took the number of recovered nodes in the network as the influence of the node selected as the single infected node at the beginning of that simulation. We repeated the simulation for each node 1000 times and took the average of their influences as the final SIR score. For the simulations we used Python and NetworkX [46].

4.3.1. Kendall Ranking

The best results were given by EVE in six experiments, by GC in two experiments, and by EC in one experiment. In addition, the EVE tau values in all experiments are very close to 0.8 or higher. The more detailed results are shown in Figure 2.

The ranking performances of EVE and the competitor centrality measures for $\beta = \beta_{th}$are shown in Table 2. The best results are emphasized in bold. Ranking performances were calculated using Definition 2, as the Kendall's tau ranking correlation coefficient. The ranking list created by the measure and the list created by SIR simulations were used in the calculations.

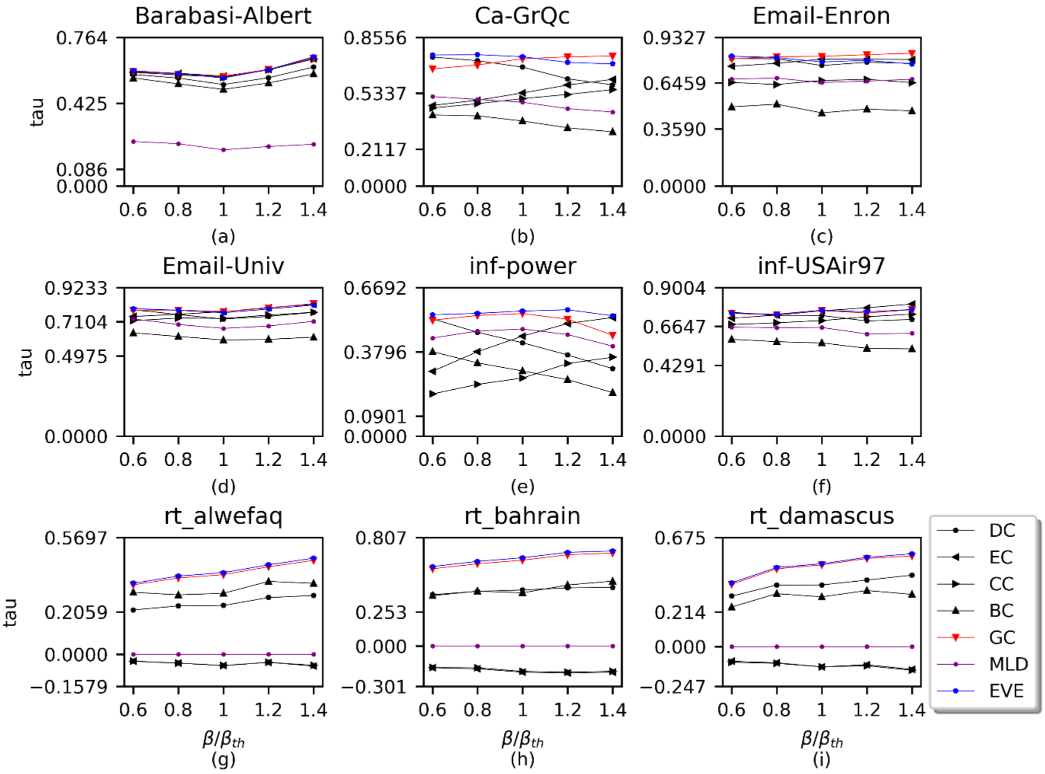


Figure 2. Kendall's tau correlation coefficient results of the centrality measures.

Table 2. Kendall's tau correlation coefficient results of the centrality measures for $\beta = \beta_{th}$.

Dataset	DC	EC	CC	BC	GC	MLD	EVE
Barabasi-Albert	0.5214	**0.5645**	0.5595	0.4965	0.5622	0.1859	0.5550
Ca-GrQc	0.6840	0.5353	0.5030	0.3740	0.7322	0.4822	**0.7439**
Email-Enron	0.7559	0.7959	0.6617	0.4590	**0.8137**	0.6490	0.7781
Email-Univ	0.7271	0.7734	0.7310	0.5975	**0.7737**	0.6691	0.7663
inf-power	0.4204	0.4504	0.2620	0.2936	0.5519	0.4825	**0.5631**
inf-USAir97	0.7307	0.7592	0.6998	0.5649	0.7608	0.6582	**0.7629**
rt_alwefaq	0.2380	-0.0553	-0.0562	0.2977	0.3888	0.0	**0.3988**
rt_bahrain	0.4174	-0.1950	-0.1893	0.3965	0.6397	0.0	**0.6577**
rt_damascus	0.3819	-0.1259	-0.1255	0.3080	0.5064	0.0	**0.5134**

4.3.2. Ranking Monotonicity

The monotonicity values of the ranking lists created by EVE and the competitor centrality measures are shown in Table 3. The values were calculated using Definition 3. Since the ranking lists created by the centrality measures depend only on the network structure, their monotonicity values were calculated only once for each data set. The ranking list created by EVE is dependent on β. So, its monotony values should be calculated for each β. On the other hand, the monotonicity of the EVE at different β values are very close to the monotonicity of the EVE at $\beta = \beta_{th}$. For the sake of brevity, we only give the monotonicity of the EVE values at $\beta = \beta_{th}$. The monotonicity values calculated for EVE are 1 in three experiments and very close to 1 in the other three experiments. Meanwhile, the EC, CC, GC, and MLD also yielded successful results. GC and EVE have given competitor results for the retweet networks (rt_alwefaq, rt_bahrain, rt_damascus).

Finally, we examined how many of the nodes in the top x% of the ranking lists created by the measure coincided with the nodes in the top x% of the ranking list created according to the SIR simulations. The results are shown in Tables 4-5. The best results are emphasized in bold. Nodes in the top-rank levels formed by the measure are expected to be more influential nodes. Therefore, the nodes at the top of the list and those at the top of the ranking list created according to the SIR simulations must be the same. According to the results, EVE outperforms the competitors in four experiments for top 3% and top5% of the ranking lists.

Table 3. Monotonicity values of the centrality measures.

Dataset	DC	EC	CC	BC	GC	MLD	EVE
Barabasi-Albert	0.9279	1.0	0.9999	1.0	1.0	1.0	1.0
Ca-GrQc	0.9647	0.9973	1.0	0.7892	1.0	1.0	1.0
Email-Enron	0.9958	1.0	1.0	0.9998	1.0	1.0	1.0
Email-Univ	0.97316	0.9995	0.9995	0.9920	0.9995	0.9995	0.9995
inf-power	0.8043	0.6897	0.9947	0.9543	0.9939	0.9947	0.9947
inf-USAir97	0.9830	0.9966	0.9944	0.9403	0.9961	0.9956	0.9956
rt_alwefaq	0.6933	0.9672	0.9944	0.0878	0.2680	0.0698	0.2635
rt_bahrain	0.7677	0.8805	0.9910	0.1590	0.4866	0.1991	0.4800
rt_damascus	0.3999	0.3130	0.9543	0.0506	0.2382	0.0979	0.2359

Table 4. The number of matching nodes in the top 3% of the ranking list was created according to SIR simulations with the ranking lists created by the measures.

Dataset	DC	EC	CC	BC	GC	MLD	EVE
Barabasi-Albert	24	25	**26**	24	25	23	24
Ca-GrQc	18	19	13	2	**21**	0	19
Email-Enron	1	**4**	1	1	3	2	1
Email-Univ	20	**24**	17	14	19	16	21
inf-power	47	**88**	16	13	86	48	50
inf-USAir97	**8**	**8**	6	4	**8**	7	**8**
rt_alwefaq	57	17	12	42	49	29	**61**
rt_bahrain	106	11	5	58	90	23	**110**
rt_damascus	34	3	4	26	36	26	**36**

Table 5. The number of matching nodes in the top 5% of the ranking list was created according to SIR simulations with the ranking lists created by the measures.

Dataset	DC	EC	CC	BC	GC	MLD	EVE
Barabasi-Albert	40	**41**	**41**	**41**	**41**	39	**41**
Ca-GrQc	36	26	20	7	**38**	0	37
Email-Enron	4	4	3	3	4	**5**	3
Email-Univ	37	36	35	31	**38**	34	36
inf-power	80	**150**	46	39	134	91	101
inf-USAir97	12	12	10	9	**13**	12	12
rt_alwefaq	77	53	43	55	68	71	**82**
rt_bahrain	161	18	6	97	158	63	**181**
rt_damascus	86	24	23	49	83	63	**93**

5. Discussion and Conclusions

In this study, we proposed an approach that approximates the influences of nodes in complex networks under the SIR propagation model using the shortest paths between nodes and then applies this to rank the nodes. The EVE is similar to a centrality measure in that it is used for ranking nodes. However, EVE is not a general centrality measure, but a metric specific to the SIR model. As a result of nine datasets and five different SIR settings, EVE performed better than state-of-the-art and well-known centrality measures.

We compared EVE with well-known centrality measures as well as with a state-of-the-art measure such as MLD, and GC, which are successful and innovative methods. The EVE demonstrated that the expected influences of nodes could be better distinguished by using the parameters of the propagation model and the shortest paths (without using the centrality measures of the nodes).

The EVE is calculated using the shortest paths between nodes. This means that all other paths are ignored. In dense networks, there can be many different paths other than the shortest path between two nodes. Therefore, ignoring these paths increase the difference (error) between EVE and the actual expected influence.

The time complexity of the EVE mostly depends on the shortest path calculation. The EVE needs to calculate all-pairs shortest path, and the known best worst-case time complexity for this purpose is $O(|V|^3)$. After the calculation of all-pairs shortest paths, the EVE algorithm works. It has two nested loops that both have $|V|$ as the upper limit. So, time complexity of this part of the algorithm is $O(|V|^2)$. Of two consecutive algorithm parts, the complexity of the larger one is the complexity of the entire algorithm. Thus, the total complexity of EVE becomes $O(|V|^3)$.

As future work, EVE-like algorithms can be developed for other propagation models. Additionally, improvements can be made to EVE to obtain more accurate results on dense networks.

Acknowledgment

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J. Gao, B. Barzel, and A.-L. Barabási, “Universal resilience patterns in complex networks,” *Nature*, vol. 530, no. 7590, pp. 307–312, Feb. 2016, doi: 10.1038/nature16948.

[2] A.-L. Barabási, N. Gulbahce, and J. Loscalzo, “Network medicine: a network-based approach to human disease,” *Nat. Rev. Genet.*, vol. 12, no. 1, pp. 56–68, Jan. 2011, doi: 10.1038/nrg2918.

[3] H. Alasmary *et al.*, “Analyzing and Detecting Emerging Internet of Things Malware: A Graph-Based Approach,” *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8977–8988, 2019, doi: 10.1109/JIOT.2019.2925929.

[4] J. Yang, C. Yao, W. Ma, and G. Chen, “A study of the spreading scheme for viral marketing based on a complex network model,” *Phys. A Stat. Mech. its Appl.*, vol. 389, no. 4, pp. 859–870, Feb. 2010, doi: 10.1016/j.physa.2009.10.034.

[5] İ. Tuğal and A. Karcı, “Comparisons of Karcı and Shannon entropies and their effects on centrality of social networks,” *Phys. A Stat. Mech. its Appl.*, vol. 523, pp. 352–363, Jun. 2019, doi:

10.1016/j.physa.2019.02.026.

[6] J. Leskovec, D. Huttenlocher, and J. Kleinberg, “Predicting positive and negative links in online social networks,” in *Proceedings of the 19th international conference on World wide web - WWW ’10*, 2010, p. 641. doi: 10.1145/1772690.1772756.

[7] S. P. Borgatti, A. Mehra, D. J. Brass, and G. Labianca, “Network Analysis in the Social Sciences,” *Science (80-.).*, vol. 323, no. 5916, pp. 892–895, Feb. 2009, doi: 10.1126/science.1165821.

[8] S. Chang *et al.*, “Mobility network models of COVID-19 explain inequities and inform reopening,” *Nature*, vol. 589, no. 7840, pp. 82–87, Jan. 2021, doi: 10.1038/s41586-020-2923-3.

[9] Y. Yang, X. Wang, Y. Chen, M. Hu, and C. Ruan, “A Novel Centrality of Influential Nodes Identification in Complex Networks,” *IEEE Access*, vol. 8, pp. 58742–58751, 2020, doi: 10.1109/ACCESS.2020.2983053.

[10] J. Zhang, C. Yang, Z. Jin, and J. Li, “Dynamics analysis of SIR epidemic model with correlation coefficients and clustering coefficient in networks,” *J. Theor. Biol.*, vol. 449, pp. 1–13, 2018, doi: 10.1016/j.jtbi.2018.04.007.

[11] S. Banerjee, M. Jenamani, and D. K. Pratihar, “A survey on influence maximization in a social network,” *Knowl. Inf. Syst.*, 2020, doi: 10.1007/s10115-020-01461-4.

[12] D. Kempe, J. Kleinberg, and É. Tardos, “Maximizing the spread of influence through a social network,” in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’03*, 2003, p. 137. doi: 10.1145/956750.956769.

[13] S. P. Borgatti, “Identifying sets of key players in a social network,” *Comput. Math. Organ. Theory*, vol. 12, no. 1, pp. 21–34, Apr. 2006, doi: 10.1007/s10588-006-7084-x.

[14] L. C. Freeman, “A Set of Measures of Centrality Based on Betweenness,” *Sociometry*, vol. 40, no. 1, p. 35, Mar. 1977, doi: 10.2307/3033543.

[15] L. Katz, “A new status index derived from sociometric analysis,” *Psychometrika*, vol. 18, no. 1, pp. 39–43, Mar. 1953, doi: 10.1007/BF02289026.

[16] L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank Citation Ranking: Bringing Order to the Web.,” Stanford InfoLab, 1999. [Online]. Available: http://ilpubs.stanford.edu:8090/422/

[17] J. Sheng *et al.*, “Identifying influential nodes in complex networks based on global and local structure,” *Phys. A Stat. Mech. its Appl.*, vol. 541, p. 123262, 2020, doi: 10.1016/j.physa.2019.123262.

[18] C. Salavati, A. Abdollahpouri, and Z. Manbari, “Ranking nodes in complex networks based on local structure and improving closeness centrality,” *Neurocomputing*, vol. 336, pp. 36–45, 2019, doi: 10.1016/j.neucom.2018.04.086.

[19] Z. Ghalmane, M. El Hassouni, C. Cherifi, and H. Cherifi, “Centrality in modular networks,” *EPJ Data Sci.*, vol. 8, no. 1, 2019, doi: 10.1140/epjds/s13688-019-0195-7.

[20] J. Zhao, Y. Song, and Y. Deng, “A novel model to identify the influential nodes: Evidence theory centrality,” *IEEE Access*, vol. 8, pp. 46773–46780, 2020, doi: 10.1109/ACCESS.2020.2978142.

[21] Z. Lv, N. Zhao, F. Xiong, and N. Chen, “A novel measure of identifying influential nodes in complex networks,” *Phys. A Stat. Mech. its Appl.*, vol. 523, pp. 488–497, 2019, doi: 10.1016/j.physa.2019.01.136.

[22] M. Curado, L. Tortosa, and J. F. Vicent, “A novel measure to identify influential nodes: Return Random Walk Gravity Centrality,” *Inf. Sci. (Ny).*, vol. 628, pp. 177–195, May 2023, doi: 10.1016/j.ins.2023.01.097.

[23] Z. Zhao, X. Wang, W. Zhang, and Z. Zhu, “A community-based approach to identifying influential spreaders,” *Entropy*, vol. 17, no. 4, pp. 2228–2252, 2015, doi: 10.3390/e17042228.

[24] Z. Ghalmane, C. Cherifi, H. Cherifi, and M. El Hassouni, “Centrality in Complex Networks with Overlapping Community Structure,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–29, 2019, doi: 10.1038/s41598-019-46507-y.

[25] Y. Zhao, S. Li, and F. Jin, “Identification of influential nodes in social networks with community structure based on label propagation,” *Neurocomputing*, vol. 210, pp. 34–44, 2016, doi: 10.1016/j.neucom.2015.11.125.

[26] Y. Y. Keng, K. H. Kwa, and C. McClain, “Convex combinations of centrality measures,” *J. Math. Sociol.*, 2020, doi: 10.1080/0022250X.2020.1765776.

[27] S. S. Ali, T. Anwar, and S. A. M. Rizvi, “A Revisit to the Infection Source Identification Problem under Classical Graph Centrality Measures,” *Online Soc. Networks Media*, vol. 17, no. xxxx, p. 100061, May

2020, doi: 10.1016/j.osnem.2020.100061.

[28] M. Alshahrani, Z. Fuxi, A. Sameh, S. Mekouar, and S. Huang, "Efficient algorithms based on centrality measures for identification of top-K influential users in social networks," *Inf. Sci. (Ny).*, vol. 527, pp. 88–107, Jul. 2020, doi: 10.1016/j.ins.2020.03.060.

[29] M. Şimşek and H. Meyerhenke, "Combined centrality measures for an improved characterization of influence spread in social networks," *J. Complex Networks*, vol. 8, no. 1, Feb. 2020, doi: 10.1093/comnet/cnz048.

[30] L. Ma, C. Ma, H. Zhang, and B. Wang, "Identifying influential spreaders in complex networks based on gravity formula," *Phys. A Stat. Mech. its Appl.*, vol. 451, pp. 205–212, Jun. 2016, doi: 10.1016/j.physa.2015.12.162.

[31] X.-L. Yan, Y.-P. Cui, and S.-J. Ni, "Identifying influential spreaders in complex networks based on entropy weight method and gravity law," *Chinese Phys. B*, vol. 29, no. 4, p. 048902, Apr. 2020, doi: 10.1088/1674-1056/ab77fe.

[32] A. Şimşek, "Lexical sorting centrality to distinguish spreading abilities of nodes in complex networks under the Susceptible-Infectious-Recovered (SIR) model," *J. King Saud Univ. - Comput. Inf. Sci.*, Jun. 2021, doi: 10.1016/j.jksuci.2021.06.010.

[33] X. Wen, C. Tu, M. Wu, and X. Jiang, "Fast ranking nodes importance in complex networks based on LS-SVM method," *Phys. A Stat. Mech. its Appl.*, vol. 506, pp. 11–23, Sep. 2018, doi: 10.1016/j.physa.2018.03.076.

[34] L. Sabah and M. Şimşek, "A new fast entropy-based method to generate composite centrality measures in complex networks," *Concurr. Comput. Pract. Exp.*, vol. 35, no. 10, May 2023, doi: 10.1002/cpe.7657.

[35] J. Liu, J. Lin, Q. Guo, and T. Zhou, "Locating influential nodes via dynamics-sensitive centrality," *Sci. Rep.*, vol. 6, no. 1, p. 21380, Feb. 2016, doi: 10.1038/srep21380.

[36] D. Tolić, K.-K. Kleineberg, and N. Antulov-Fantulin, "Simulating SIR processes on networks using weighted shortest paths," *Sci. Rep.*, vol. 8, no. 1, p. 6562, Dec. 2018, doi: 10.1038/s41598-018-24648-w.

[37] M. G. Kendall, "A New Measure of Rank Correlation," *Biometrika*, vol. 30, no. 1–2, pp. 81–93, Jun. 1938, doi: 10.1093/biomet/30.1-2.81.

[38] J. Bae and S. Kim, "Identifying and ranking influential spreaders in complex networks by neighborhood coreness," *Phys. A Stat. Mech. its Appl.*, vol. 395, pp. 549–559, Feb. 2014, doi: 10.1016/j.physa.2013.10.047.

[39] G. Rossetti, L. Milli, S. Rinzivillo, A. Sîrbu, D. Pedreschi, and F. Giannotti, "NDlib: a python library to model and analyze diffusion processes over complex networks," *Int. J. Data Sci. Anal.*, vol. 5, no. 1, pp. 61–79, Feb. 2018, doi: 10.1007/s41060-017-0086-6.

[40] M. Newman, *Networks*, Second. Oxford, UK: Oxford University Press, 2018.

[41] P. Bonacich, "Power and Centrality: A Family of Measures," *Am. J. Sociol.*, vol. 92, no. 5, pp. 1170–1182, Mar. 1987, doi: 10.1086/228631.

[42] G. Sabidussi, "The centrality index of a graph," *Psychometrika*, vol. 31, no. 4, pp. 581–603, Dec. 1966, doi: 10.1007/BF02289527.

[43] T. Wen, D. Pelusi, and Y. Deng, "Vital spreaders identification in complex networks with multi-local dimension," *Knowledge-Based Syst.*, vol. 195, p. 105717, 2020, doi: 10.1016/j.knosys.2020.105717.

[44] R. A. Rossi and N. K. Ahmed, "The Network Data Repository with Interactive Graph Analytics and Visualization," 2015. [Online]. Available: http://networkrepository.com

[45] Z. Li, T. Ren, X. Ma, S. Liu, Y. Zhang, and T. Zhou, "Identifying influential spreaders by gravity model," *Sci. Rep.*, vol. 9, no. 1, pp. 1–7, 2019, doi: 10.1038/s41598-019-44930-9.

[46] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring Network Structure, Dynamics, and Function using NetworkX," in *Proceedings of the 7th Python in Science Conference*, 2008, pp. 11–15.