2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

# Blocking the Spread of Misinformation in a Network under Distinct Cost Models

Fernando C. Erd
*Departament of Computer Science*
*Federal University of Paraná*
Curitiba, Brazil
fcerd@inf.ufpr.br

André L. Vignatti
*Departament of Computer Science*
*Federal University of Paraná*
Curitiba, Brazil
vignatti@inf.ufpr.br

Murilo V. G. da Silva
*Departament of Computer Science*
*Federal University of Paraná*
Curitiba, Brazil
murilo@inf.ufpr.br

***Abstract*—Given a network $N$ and a set of nodes that are the starting point for the spread of misinformation across $N$ and an integer $k$, in the *influence blocking maximization* problem the goal is to find $k$ nodes in $N$ as the starting point for a competing information (say, a correct information) across $N$ such that the reach of the misinformation is minimized. In this paper we deal with a more realistic scenario for this problem where different nodes have different costs and the counter strategy has a "budget" for picking nodes for a solution. Our experimental results show that the success of a given strategy varies substantially depending on the cost function in the model. In particular, we investigate the cost function where all nodes have cost 1 and a cost function that assigns higher costs to higher degree nodes. We show that, even though strategies that perform well in these two diverse cases are very different from each other, both correlate well with simple (but different) strategies: greedily choose high degree nodes and choose nodes uniformly at random.**

***Index Terms*—influence blocking maximization, misinformation, complex networks.**

## I. Introduction

The spread of misinformation is not a new phenomenon, however, with the prevalence of social media this problem seems to have being gaining more momentum [1]. There are evidence that people tend to believe in information that matches their perception of social narratives and to discredit narratives that deconstruct that perception [2].

Some studies showed that the spread of misinformation has potential to influence the behavior of the society. Allcott and Gentzkow (2017) [3] present an analysis of how misinformation may have affected the result of the 2016 United States elections. Another example is the number of questionable sources on the main social platforms regarding the outbreak of COVID-19, as shown by Cinelli et al. (2020) [4].

The algorithmic aspects of a problem originally from the field of "viral marketing" was investigated by Kempe (2003) [5]. The proposed computational problem, known as *influence maximization* in networks, is the following. Given a network, the goal is to select the best individuals to advertise a product, such that the information about that product reaches the largest number of people. From this problem, a line of research arose addressing the problem of finding a counter strategy for the spread of such influence [6]. In our paper we assume that we are dealing with the spread of misinformation and that the counter strategy seeks to spread the correct information. This computational problem, called *influence blocking maximization*, is the following. Given a set of nodes as starting point for the spread of misinformation across the network and an integer $k$, the goal is to find $k$ nodes for the spread of a correct information across the network so the reach of the misinformation is minimized.

In the previous work in this field [6], [7], [8], [9], (we discuss these works in detail in Section II) given $k$, the counter strategy is able to pick any set of nodes of size $k$ for blocking the misinformation. We note that this scenario might be unrealistic, since choosing a node with very high degree might be much more expensive than a node of degree one, for example. In fact, In all previous works using models based on the independent cascade, the proposed strategies for choosing the set of $k$ nodes for the counter strategy perform only marginally better than choosing nodes of high degree (the algorithms are about 1% more effective than picking high degree nodes). So, in our work we generalize the problem so that different nodes might have different costs and the counter strategy has a "budget" $k$ for finding a set of nodes such that the total cost of the nodes in the set stays within that budget.

In our paper we investigate counter strategies in this generalized scenario using two distinct cost functions. Our experimental results show that the success of a given strategy varies substantially depending on the cost function in the model. The counter strategies used in our experiments are four node properties well-known in the literature: betweenness centrality, percolation centrality, PageRank and clustering coefficient. The two different cost functions that we compare are the *uniform* cost function and *degree penalty* cost function. This second cost function may be a more realistic since nodes of high degree in a network might be more expensive. In consonance with previous results, we show that for the uniform cost function a number of winning strategies correlate well with simply choosing high degree nodes. In the degree penalty cost function, we show that the same does not hold. Interestingly, we show that there is also a simple strategy in this scenario: picking nodes uniformly at random for the solution.

The rest of this article is organized as follows: a brief review of recent research in the field is provided in Section

IEEE/ACM ASONAM 2020, December 7-10, 2020
978-1-7281-1056-1/20/$31.00 © 2020 IEEE

II. Section III presents the MCICM information dissemination model. The problem definition is described in Section IV. The methodology used for our results is discussed in Section V. Experimental results on some well-known data sets are reported in Section VI. Finally, Section VII concludes the work.

## II. Related Work

The influence maximization problem [5] in a network is the computational problem of finding a set of nodes of size $k$, for a given integer $k$ that is part of the input, as the starting point for the spread of information in this network so that the maximum number of nodes is reached. In this paper we deal with a version of the influence maximization problem where there are two competing information being disseminated in the network, referred here as the *misinformation* and the *correct information*. Such competitive version of the influence maximization problem is formally proposed by He et al. (2012) [6]. In this scenario the input consists of a network with $k$ given nodes for spreading the misinformation and the goal is to find $k$ nodes for spreading the correct information so that the number of nodes reached by the misinformation is minimized. Using a variation of the linear threshold model, called competitive linear threshold model (CLT), the authors show that the problem is submodular and monotonic, which guarantees an approximation of $1-1/e$ of the optimal solution using a hill climbing strategy. Also, they propose the CLDAG algorithm, based on the LDAG [10] algorithm which was previously used for the influence maximization problem.

The first work in this context using the independent cascade model in the competitive version appeared in Budak et al. (2009) [11]. In this paper, the authors proposed the *eventual influence limitation problem* (EIL), where the cascade of negative (false) information propagates alone in the network for a certain number of steps, before the cascade of positive information starts to spread through the network. Budak's main contribution is a proof that the function associated to the problem is submodular and monotonic over the campaign-oblivious independent cascade (COICM) model. In addition, Budak showed that using the multi-campaign independent cascade model (MCICM) when the probabilities of positive and negative dissemination are arbitrary, the submodularity property does not hold, but when the probability of positive dissemination is equal to 1 for all edges, then the model can guarantee an approximation to the optimal solution.

In the MCICM model, Arazkhani et al. (2019) [7] used a metric based on some centrality measures, like degree, betweenness and closeness, in order to choose the set of positive seed nodes. In a later study Arazkhani et al. (2019) [8] combined the centralities in a pre-processing method to find the largest $k$ communities using fuzzy clustering, which chooses a node with the highest degree, betweenness or closeness of each community as being the positive seed. Regarding the dissemination taking place on the COICM model, Wu et al. (2017) [9] used the structure of maximum influence arborescence (MIA) proposing two heuristics, CMIA-H and CMIA-O. In the same work they consider the MCICM model in the particular case where the probability of positive dissemination is 1 for all edges.

A variant of the influence maximization problem proposed by Kempe et al. [5] considers costs for selecting each node in the network. For example, the *budgeted influence maximization* problem studied by [12], each node $v$ is associated with an arbitrary cost $c(v)$. The goal of such problem is to select a set $S$ of nodes so that the cost of those nodes in $S$ is at most a budget $b$, and $S$ maximizes the spread of information thought the network. In the *budgeted competitive influence maximization* problem, proposed by [13], the goal is to maximize the spread of one product over another, given a budget.

## III. Diffusion Model

In this work, we use the *multi-campaign independent cascade model* (MCICM) introduced by Budak et al. [11]. In MCICM, given a directed or undirected graph $G = (V, E)$ there are two spreading cascades $P$ and $N$ representing the positive and negative cascades respectively, two initial sets $S \subseteq V$ and $N_0 \subseteq V$ of positive and negative seeds respectively. The negative seeds are the starting point for the misinformation and the positive seeds the starting point for the correct information. Each node assumes three different states: positive, negative or inactive, and in the starting configuration the nodes in $S$ are set as positive, those in $N_0$ are set as negative and the rest of the nodes are set as inactive. In addition, each edge $(u, v) \in E$ has two weights, $w^+_{u,v}$ and $w^-_{u,v}$ in the range $[0, 1]$, which denote the probabilities of $u$ activating, respectively, positively or negatively the node $v$. The simulation occurs in discrete time steps, and if $u$ is activated in step $t$ by the cascade of $P$ or $N$, it has only one chance to positively or negatively activate a neighbor $v$ during the simulation. As a tiebreaker rule, if the $P$ cascade and the $N$ cascade in the same step $t$ try to activate the same inactive node, the $N$ cascade has preference for the activation. The step $t$ finish when all nodes activated during step $t-1$ try to activate their inactive neighbors and simulation ends in step $t$ when no node is activated by the cascades.

## IV. Problem Definition

Let $N_T$ be the set of negative nodes that is the outcome of an execution of the stochastic process of diffusion (in our case, dictated by the MCICM model). The outcome $N_T$ depends on the graph $G$, the probabilities $w^+$ and $w^-$, and the initial negative and positive seed sets $N_0$ and $S$. Thus, given an integer $k$, the probability that $|N_T| = k$ depends on the same input variables, but in the notation we only make it explicit the dependence on the initial positive seed set $S$, writing $\Pr\big(|N_T| = k \,\big|\, S\big)$.

Now, given the initial positive seed set $S$, the expected size of the negative nodes $N_T$ is,

$$\mathbb{E}\big[|N_T| \;\big|\; S\big] = \sum_{k=0}^{|V|} k \cdot \Pr\big(|N_T| = k \,\big|\, S\big).$$

We can measure the impact of an initial positive seed set $S$ by considering the difference between two scenarios, when the initial positive seed set is $S$, and when the initial positive seed set is empty. This is called the *expected blocked negative influence* of $S$, and is formally defined as

$$\sigma(S) = \mathbb{E}\big[|N_T| \;\big|\; \{\emptyset\}\big] - \mathbb{E}\big[|N_T| \;\big|\; S\big],$$

and we want to maximize this quantity. We can now define the problem.

**Problem 1** (Generalized Influence Block Maximization (GIBM))**.** *Given a graph $G = (V, E)$ with costs $c(v)$ for each $v \in V$, propagation probabilities $w^+$ and $w^-$, a negative seed set $N_0$, and a positive integer $k$, the* GIBM problem *aims to find the positive seed set $S$ that maximizes $\sigma(S)$ where $\sum_{v \in S} c(v) \leq k$.*

## V. Methodology

Our goal is to investigate several measures of centrality to be used as strategies to solve the problem under the *uniform* cost function, where all nodes have the same cost, w.l.o.g., say, $c(v) = 1$ and *degree penalty* cost function, such that $c(v) = \delta(v)$, where $\delta(v)$ be the degree of a node $v$. For both cost functions cases, we perform simulations on directed and undirected graphs, with the goal of analyzing the behavior between the two types of graphs.

If the input graph is not connected, we take into consideration only the largest connected component (resp. largest weakly connected component for directed graphs) of the graph. In the simulations for each edge $w^+$ and $w^-$ are chosen in the interval $[0, 1]$ so that independently and randomly from the uniform distribution. Various sizes of $N_0$ are considered in the experiments.

For the experiments we choose three real-world datasets, among which are two networks of citations (DBLP and CORA) and the Wikipedia Election dataset. The original DBLP database has 12,590 nodes and 49,749 edges. The CORA dataset [14] contains more than 23,000 nodes and approximately 90,000 edges. The Wikipedia dataset [15] represents the English Wikipedia social network and has 7,066 nodes and more than 100,000 edges. All datasets originally describe directed graphs. The same datasets were used in the experiments on undirected graphs, but the direction of the edges was ignored.

For each proposed scenario we use the following network metrics as a counter strategy: clustering coefficient [16], PageRank [16], betweenness [16] and percolation [17]. In addition to the measures above, we use two strategies for experiment control: choosing high degree nodes first (greedly) and choosing nodes at random.

The percolation centrality requires weights for nodes reflecting a certain degree of "contamination", so we use this measure in our experiments in the following way. Let $d(v, N_0)$ be the distance from $v$ to the nearest node in $N_0$. Thus, the percolation weight for a node $v$ is defined as

$$\text{perc}(v) = \frac{1}{d(v, N_0) + 1}.$$

The idea is that the nodes initially in $N_0$ are 100% percolated (in this case, $d(v, N_0) = 0$), and as a node is further away from $N_0$, its percolation weight decreases.

The experiments were launched in an Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz and 8 GB RAM. The scripts were implemented in Python 3.6.9 language. For graph manipulations we use the NetworkX 2.3 library [18]. The implementation of all the networks measures considered in this work are available in NetworkX as well.

## VI. Experiments and Results

In this section we evaluate the performance of different strategies for finding a solution for the GIBM problem. Since MCICM is a probabilistic model, we run repeated experiments for the spreading over the initial sets $N_0$ and $S$ in order to obtain the average behavior. In each different scenario, we perform the simulation 1000 times to obtain the average of the positively and negatively contaminated sets.

### A. Uniform Cost Function

In this section we show and analyze the results obtained for the *uniform* cost function. For these experiments, we set the size of $N_0$ to be 1% of the number of nodes of each dataset, and we vary the parameter $k$ (here the size of the output set $S$ for positive seeds equals $k$) between 0.1%, 0.5%, 1%, 1.5% and 2.0% of the number of nodes in each dataset. We analyze the undirected and directed cases. In each plot, we show the average results for the three datasets. The vertical axis we show the percentage of negatively contaminated nodes, therefore, the lower the values, the better the network metric works as a strategy for the problem.

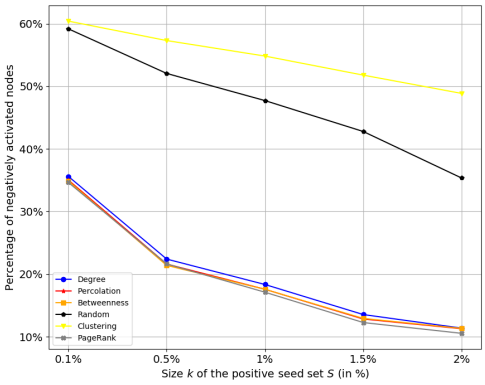


Fig. 1. *Uniform* cost function in undirected graphs.

In the undirected graphs case, show in Figure 1, we see that the percolation, betweenness, degree and PageRank measures behave similarly, and also they perform better than other strategies. In the case of directed graphs, the number of positively influenced nodes decreases in comparison to the undirected version, as show in Figure 2.

We hypothesize that the similar behaviors between degree, betweenness, PageRank and percolation comes from the fact

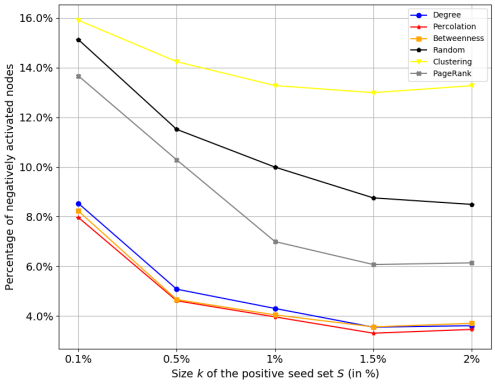


Fig. 2. *Uniform* cost function in directed graph.

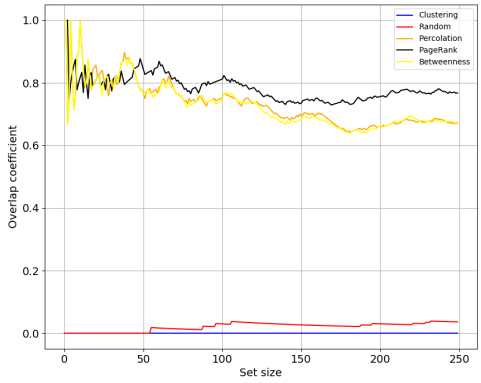


Fig. 3. Overlap coefficient in DBLP undirected graph: value 0 (resp. value 1) is the case where the elements of the solution are completely different (resp. exactly the same) from nodes of $k$ highest degrees.

that the set of positive seeds chosen by these strategies are similar. In order to test this hypothesis, we take the set of positive seeds of the degree centrality as a basis for the comparison, and measure the similarity between the sets returned by the other strategies. More formally, let $S_1$ and $S_2$ be the sets returned by using, respectively, the degree centrality and some other strategy. To measure the similarity between the sets, we use the *overlap coefficient*, defined as

$$\frac{|S_1 \cap S_2|}{\min\{|S_1|, |S_2|\}}.$$

Figure 3 shows the results of the similarities between the solutions on the DBLP dataset. On the vertical axis we have the overlap coefficient, taking the degree centrality as the base comparison. The horizontal axis represents the size of the set, and we show solutions up to 250 nodes since this is roughly the size of the largest sets for the solutions in the experiments and, additionally, with the solution size approaching the entire node set obviously they have a large overlap. We note that solutions using betweenness, percolation and PageRank as strategies have a high overlap coefficient. This means that the solution sets returned by these strategies are similar to the degree centrality strategy. On the other hand, solutions obtained using clustering coefficient and random sampling as strategy have a very small overlap, so they are very different from the set nodes with highest degree.

### B. Degree Penalty Cost Function

In this section we analyze the results for the *degree penalty* cost function. In this case the costs are directly proportional to the degree, so we define the sizes of $N_0$ and $S$ as a fraction of the sum of the degrees (i.e., twice the number of edges). More specifically, we set the size of $N_0$ to be equal to $1\%$ of the sum of the degrees and choose $k$ to be 0.1%, 0.5%, 1%, 1.5% and 2% of that same sum.

Initially, we analyze the results for undirected graphs. Differently from the case with uniform cost function where the node degree is the central attribute that characterize the success of a given strategy, in the degree penalty cost function the

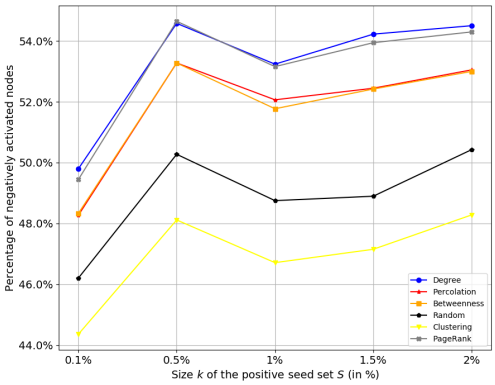


Fig. 4. *Degree penalty* cost function in undirected graph.

node weight "amortizes" the advantage that the degree exerts in those strategies where high degree nodes are prioritized, i.e., betweenness, percolation and PageRank (recall Figure 3 where we show the overlap of such strategies with the set of highest degree nodes). Therefore, these strategies are not as successful in the scenario using the *degree penalty* cost function as shown in Figure 4. Generally speaking, compared to the *uniform* cost function, the *degree penalty* cost function had more negatively influenced nodes. Also, in the *degree penalty* cost function problem, the clustering and random strategies present the best performances among the metrics we choose.

In particular, we believe that the good performance of the clustering coefficient can be explained by the dissimilarity between this measure and the degree centrality (also shown in Figure 3). In the analyzed datasets, the nodes with the highest clustering coefficient are those with the lowest degree. Since the strategy that uses the clustering coefficient selects nodes with low degree, this means that it chooses a large number of nodes for the solution, as the cost of the nodes in this case is low. Therefore, the clustering coefficient strategy may succeed

by being able to choose a high fraction of the nodes of a graph.

The random strategy also had good results and we have some supposition for its success. Since, in real-world graphs, typically, the degree distribution is approximated by a power law distribution, roughly speaking theses graphs contain a large number of low degree nodes. This may explain, in part, the good performance of the random metric. The idea is that by randomly selecting the graph vertices, the vast majority are low degree vertices and therefore more vertices are selected until reaching the maximum budget limit.

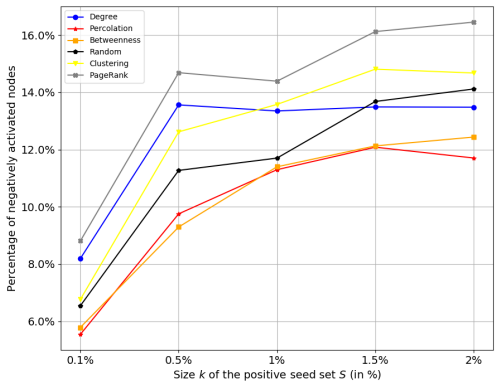


Fig. 5. *Degree penalty* cost function in directed graphs.

The behavior of *degree penalty* cost function in directed graphs is different from the other cases analyzed so far. Figure 5 shows that betweenness and percolation metrics show better results than the other metrics. The clustering coefficient strategy had opposite performances in the directed and undirected cases. A possible explanation is that in the directed case, the nodes chosen by this strategy have low degree. Thus, due to the edge directions, many may have out-degree equal to zero, making the spreading impossible.

## VII. Conclusion

In this work, we present the Generalized Influence Blocking Maximization (GIBM) problem and analyze the behavior of strategies based on well-known network metrics for two particular cost functions: *uniform* and *degree penalty*. The *uniform* cost function case has appeared in the literature as the influence blocking maximization problem. For this case, the betweenness, percolation and PageRank metrics obtain similar results to the simple degree centrality. We show that this similarity is related to the overlapping of the solution sets. On the other hand, in the *degree penalty* case, the results show that the same metrics have opposite performances. In addition, our results suggest at least two conclusions for algorithms that have a high level of similarity with the node degree. First, however sophisticated is the metric computed by an algorithm in the *uniform* cost function scenario, if there is a high similarity between such metric and the node degree, then one should not expect substantial improvements in their performance. So this might be the case why recent results in the literature obtained only slight improvements (about 1% better) when compared with the node degree strategy in the uniform cost function case. Second, algorithms with solutions correlated to the set of high degree nodes does not perform well in degree penalty scenario. This naturally leads us to consider future research where the goal is to design solutions that take into consideration other cost functions for the generalized version of the problem.

## References


[1] D. M. J. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, M. Schudson, S. A. Sloman, C. R. Sunstein, E. A. Thorson, D. J. Watts, and J. L. Zittrain, "The science of fake news," *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.

[2] S. Lewandowsky, U. K. H. Ecker, C. M. Seifert, N. Schwarz, and J. Cook, "Misinformation and its correction: Continued influence and successful debiasing," *Psychological Science in the Public Interest*, vol. 13, no. 3, pp. 106–131, 2012, pMID: 26173286.

[3] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, pp. 211–236, 2017.

[4] M. Cinelli, W. Quattrociocchi, A. Galeazzi, C. M. Valensise, E. Brugnoli, A. L. Schmidt, P. Zola, F. Zollo, and A. Scala, "The covid-19 social media infodemic," *ArXiv*, vol. abs/2003.05004, 2020.

[5] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," *Proc. of the ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, vol. 137-146, 2003.

[6] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model technical report," *Proceedings of the 12th SIAM International Conference on Data Mining, SDM*, 2012.

[7] N. Arazkhani, M. R. Meybodi, and A. Rezvanian, "Influence blocking maximization in social network using centrality measures," in *5th Conf. on Knowledge Based Eng. and Innovation (KBEI)*, 2019, pp. 492–497.

[8] N. Arazkhani, M. R. Meybodi, and A. Rezvanian, "An efficient algorithm for influence blocking maximization based on community detection," in *5th Int. Conf. on Web Research (ICWR)*, 2019, pp. 258–263.

[9] P. Wu and L. Pan, "Scalable influence blocking maximization in social networks under competitive independent cascade models," *Computer Networks*, vol. 123, 2017.

[10] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 88–97, 2010.

[11] C. Budak, D. Agrawal, and A. Abbadi, "Limiting the spread of misinformation in social networks," *Proceedings of the 20th International Conference on World Wide Web, WWW 2011*, pp. 665–674, 2011.

[12] H. Nguyen and R. Zheng, "Influence spread in large-scale social networks - A belief propagation approach," *CoRR*, vol. abs/1204.4491, 2012.

[13] C. V. Pham, H. V. Duong, H. X. Hoang, and M. T. Thai, "Competitive influence maximization within time and budget constraints in online social networks: An algorithmic approach," *Applied Sciences*, vol. 9, no. 11, 2019.

[14] L. Šubelj and M. Bajec, "Model of complex networks based on citation dynamics," in *Proceedings of the WWW Workshop on Large Scale Network Analysis*, 2013, pp. 527–530.

[15] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Governance in social media: A case study of the Wikipedia promotion process," in *Proc. Int. Conf. on Weblogs and Social Media*, 2010.

[16] M. Newman, *Networks: An Introduction*. OUP Oxford, 2010.

[17] M. Piraveenan, M. Prokopenko, and L. Hossain, "Percolation centrality: Quantifying graph-theoretic impact of nodes during percolation in networks," *PLOS ONE*, vol. 8, no. 1, pp. 1–14, 2013.

[18] 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.