

# Identification of Most Influential Spreaders in Twitter Social Network using Modified K Core Decomposition in Distributed Environment

## ABSTRACT

A clear and well-documented L<sup>A</sup>T<sub>E</sub>X document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the “acmart” document class, this article presents and explains many of the common variations, as well as many of the formatting elements an author may use in the preparation of the documentation of their work.

## CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

## KEYWORDS

datasets, neural networks, gaze detection, text tagging

### ACM Reference Format:

. 2018. Identification of Most Influential Spreaders in Twitter Social Network using Modified K Core Decomposition in Distributed Environment. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/1122445.1122456>

## 1 INTRODUCTION

Social Networks have gained remarkable popularity in the past few decades. A huge number of people are using social networking sites like Facebook, Twitter, Google+, LinkedIn etc. The heavy reliance on social networking sites causes them to generate massive data. Online social networks play a major role in the diffusion of information [5]. Since social network has great influence on society, the recent focus is on extracting valuable information from this huge amount of data. Events, issues, interests etc. happen and evolve very quickly in social networks and their capture, understanding, visualization, and prediction are becoming critical expectations from both end-users and researchers. Therefore researchers in recent years have developed a variety of techniques and models to capture information diffusion in online social networks, analyze it, extract knowledge from it and predict it [2, 16].

Micro-blogging is an emerging form of communication. One of the most notable micro-blogging services is *Twitter*. It allows *twitterers* to publish tweets (with a limit of 140 characters). *Twitter* employs a social-networking model called “following”, in which

each *twitterer* is allowed to choose who she wants to follow without seeking any permission. Conversely, she may also be followed by others without granting permission first. In one instance of “following” relationship, the *twitterer* whose updates are being followed is called the “friend”, while the one who is following is called the “follower”. *Twitter* has gained huge popularity since the first day that it was launched [43]. It has also drawn increasing interests from research community [10]. Many analysis have been done on the dataset of twitter including identification of the most influential spreader [49, 57]. We discuss the state-of-art in details in the next section.

Information diffusion is a vast research domain and has attracted research interests from many fields, such as physics, biology, etc. The diffusion of innovation over a network is one of the original reasons for studying networks and the spread of disease among a population has been studied for centuries [4, 23, 26]. The knowledge of the spreading pathways through the network of social interactions is crucial for developing efficient methods to either hinder spreading in the case of diseases, or accelerate diffusing in the case of information dissemination. The information diffusion model is based on three fundamental questions: (i) *Which pieces of information diffuse the most*, (ii) *How, why and through which paths information is diffusing*, and will be diffused in the future, (iii) *Which members of the network play important roles in the spreading process?*

In this paper we are mainly focusing on the third question. Identifying the most influential spreaders in a network is critical for ensuring efficient diffusion of information, which allows to control the outbreak of any kind of epidemic [11, 51], utilize the limited resources to optimize the information propagation [20], successful e-commercial advertisements [32, 38], optimize the use of limited resources to facilitate information propagation [9] etc. In large social network graph the nodes having the largest degree are often considered as the most influential spreader [50]. However, recent studies have shown that the most connected people are not the most influential spreader. Kitsak et al. [27] showed that the best spreaders are not necessarily the most connected people in the network. They found that the most efficient spreaders are those located within the core of the network as identified by the *k*-core decomposition analysis [55]. Moreover, many centrality measures have been proposed to identify the most influential spreaders on a social network. All these metrics are mainly of two kinds: global and local metrics. Local metrics like degree centrality and *K*-core decomposition are of simple complexity but are less effective i.e. they fail to find the most influential spreaders accurately since of omitting the global structure of the network. Global metrics such as betweenness centrality and closeness centrality perform well in the identification of the influential nodes. However, these global measures incur high computational complexity which make them

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

Woodstock '18, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/10.1145/1122445.1122456>

infeasible to use in case of large graph [19, 46]. The state-of-the-art of different methods of finding most significant nodes in a network has been discussed in the paper of Lü et al. [37] reviewed the state of the art of different proposed methods and approaches dealing with detection of vital nodes in complex networks. Moreover, available methods are compared based on the different nature of the network.

Liu et al. [35] provided an improved method of using k-shell decomposition by removing the redundant links which have a low diffusion significance. Another improved method of k-shell decomposition was proposed by Wang et al. [56] who used k-shell iteration factor to evaluate the influence capability of a node. They used the iteration information of k-shell decomposition to distinguish among the nodes with the same maximum k-shell value.

Node of the existing methods take the user info into consideration while identifying the most influential spreaders which obviously play vital role for information spreading. On top of that, with the ubiquitous use of social network, social networks are producing large amount of data. The higher the number of nodes and their friend-follower relationship data is available, the more the graph becomes suitable to identify the real most influential spreaders. However, we need special kind of

We propose a variant of *k*-core decomposition which is distributed and also incorporates the node information. Summarizing, our main contributions are:

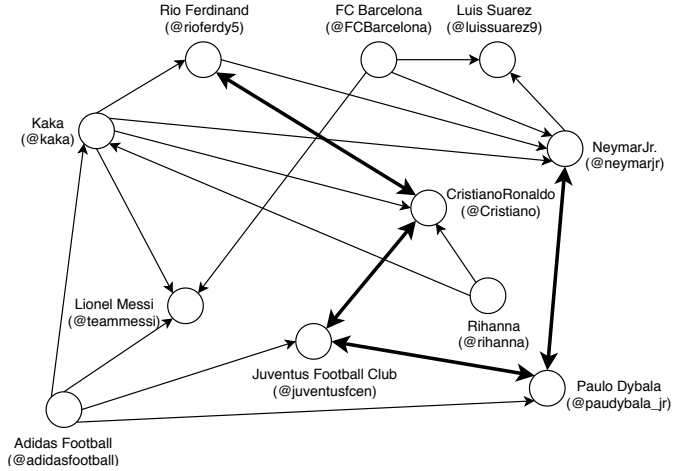
•

## 2 TECHNICAL BACKGROUND

### 2.1 Formal Definition of Twitter Social Network

The users of Twitter form a social network with their friend-follower relationship. There are several approaches to define any network formally. However, the mostly used and flexible definition can be formalized using the graph theory. That means the social network is conceptualized as a graph, i.e. a set of vertices (or nodes) representing the users of the social network and a set of edges (directed or undirected) representing social relationship between the incident vertices (users). In case of Twitter social media, we define the following two graphs:

- **Friend-Follower graph:** which is created from the relationship between the users. In this graph, a vertex represents a user and there exists a directed edge from vertex  $u$  to vertex  $v$  if and only if the user represented by  $v$  is a follower of the user represented by  $u$ . This also implies that the user represented by  $u$  is a friend of the user represented by  $v$ . For simplicity, in several cases, these directed edges may be considered as undirected. Figure 1 shows a sample friend-follower graph of twitter social network. Here every node is labeled with the name of the user along with the twitter screen name. The bidirectional edges are shown in bold edges and they implies that the users represented by the incident vertices are following one another.
- **Tweet Graph:** which is created from the retweets. In this graph, a vertex represents a user and there exists an edge between two vertices if and only if one user retweets at least



**Figure 1: sample Friend-Follower Graph of Twitter Social Network**

one of the other user's tweets. The edge should be directed to the retweeter from the user of the original tweet. For simplicity we may consider the edges as undirected.

### 2.2 Centrality Measurement

In graph theory, centrality is a term to describe importance of individual vertex within a graph or a network. The concept of centrality was first developed with an intention of doing social network analysis and there has been a lot of research works carried out in this topic for network analysis mainly to answer the question, "Which vertices are the most influential in a graph?" There are several centrality measures which are very popular in graph analysis. These measures can be categorized in two types: local and global metrics. In this section, we briefly describe various local and global centrality metrics that are available to find the influential nodes from a graph network.

**2.2.1 Local Centrality Measures.** In general, local centrality measures use only the features of an individual node through the partial information around it. The number of neighbors (degree of a node) plays the main role in such local methods and they work better mainly for undirected networks. There are mainly two local centrality measures and they are described below:

- **Degree Centrality:** Degree Centrality is the simplest centrality measure which assume that the node with the maximum neighbors possess the maximum influence on the network. The degree of the node  $v_i$  signifies the total number of edges incident to it i.e. the number of neighbors. Assume that, a graph  $G(V, E)$  where  $V$  is a set of  $n$  nodes and  $E$  is the edge set, is represented by an  $n \times n$  adjacency matrix  $A = \{a_{ij}\}$ , that is  $\{a_{ij}\} = 1$  if nodes  $v_i$  and  $v_j$  are connected and 0 otherwise. If the degree of node  $v_i$  is denoted by  $d_i$ , then the degree centrality of node  $v_i$  is:

$$DC(v_i) = \frac{d_i}{n-1} \quad (1)$$

Here  $n - 1$  is the largest possible degree in  $G$  and it is used as the numerator of the formula for normalization.

The most important side of using degree centrality to find the most influential spreader on a network is its simplicity and low computational complexity. However, in most of the cases this measure fails to identify the most influential spreaders accurately. However, there are several use cases where degree centrality can provide surprisingly good performance such as with very small spreading rate, degree centrality is a better metric to identify the spreading influences of nodes than other well-known centrality metrics [28, 34].

- **K Core Decomposition:** Degree centrality only considers the number of the adjacent neighbors in order to assess the influence of a node in the network. However, Kitsak et al. [27] identified that the location of a node in a network has a more significant aspect in evaluating its spreading influence. Nodes located at the core of a network are more likely to have a higher influence rate than those located at the periphery. Therefore, they suggested that core value of a node is a better metric in finding most influential spreaders and the core value can be obtained by using the k-core (aka k-shell) decomposition [17] of the networks.

The k-shell method starts by removing all nodes having degree 1. The process is repeated until there is no node of degree 1 exists in the network. These pruned nodes are assigned into the 1-shell. After assigning the 1-shell, all nodes with residual degree 2 are recursively removed and the 2-shell are created. This procedure continues as the residual degree increases until all nodes in the nodes have been assigned to one of the shells. The nodes with high k-shell value tend to locate in the center of the network and the spreading starting from each of these nodes are likely to widely cover the network. In this way all nodes are assigned a  $k$  (sometimes refereed to as  $k_s$ ) value. Figure ?? presents a sample network and the  $k_s$  value for the nodes by this algorithm. The algorithm is very simple and robust.

However, k core decomposition method has a tendency to assign the same  $k_s$  value to multiple nodes in case of large networks. Therefore, the hypothesis of declaring the node(s) of the largest  $k_s$  value i.e. the node(s) of the innermost shell to be the most influential results in a good number of nodes with the same spreadabilty and that may not be the desired outcome in many cases. However, the simplicity and lower computational complexity make this metric very useful to find the most influential nodes on a network and in this paper, we extend this idea and propose a modified k core decomposition method that can accurately find the most influential spreaders on a social network with very lower computational complexity.

**2.2.2 Global Centrality Measures.** Global centrality measures consider the whole network during its computation. There exists many different types of global centrality measures and each of them addresses slightly different properties of the network and the nodes in order to compute the centrality value. Two mostly used global centrality measures are closeness and betweenness centralities and they are described briefly below:

- **Closeness Centrality:** As the name suggests, closeness centrality measures how close a node is from all other nodes in a network. In case of a connected graph, the normalized closeness centrality of a node is the average length of the shortest path between the node and all other nodes in the network. Let  $d_{ij}$  be the length of shortest path between node  $v_i$  and  $v_j$ ,  $n$  be the number of nodes of the network, then the average shortest distance of node  $v_i$  will be [54],

$$Avg_i = \frac{1}{n-1} \sum_{i \neq j} d_{ij} \quad (2)$$

Since the centrality measure is intended to find the most closer nodes, the closeness centrality of node  $v_i$  is inversely proportional to the average shortest distance,  $Avg_i$  and can be defined as,

$$CC(v_i) = \frac{n-1}{\sum_{i \neq j} d_{ij}} \quad (3)$$

However, this equation will not be suitable to use in case disconnected graph where some node may be unreachable from the considering node  $v_i$ . Wasserman and Faust [47] proposed an improved formula for graphs with more than one connected component. The result is "a ratio of the fraction of nodes in the network which are reachable, to the average distance" from the reachable nodes. Let  $n_r$  be the number of reachable nodes in the network from node  $v_i$ , then the modified formula of measuring closeness centrality is,

$$CC(v_i) = \frac{n_1-1}{n-1} \frac{n_1-1}{\sum_{i \neq j} d_{ij}} \quad (4)$$

- **Betweenness Centrality:** Betweenness centrality determines how many times a node falls along the shortest path between two different nodes i.e. acts as a bridge between those two nodes. Linton Freeman [20] introduces this measure for quantifying the control of a human on the communication between other humans in a social network. For starting node  $v_s$  and destination node  $v_t$  and the input node  $v_i$  that holds the condition  $v_s \neq v_t \neq v_i$ , let  $n_{st}^i$  be 1 if node  $v_i$  lies on the shortest path between  $v_s$  and  $v_t$ ; and 0 otherwise. So the betweenness centrality is defined as:

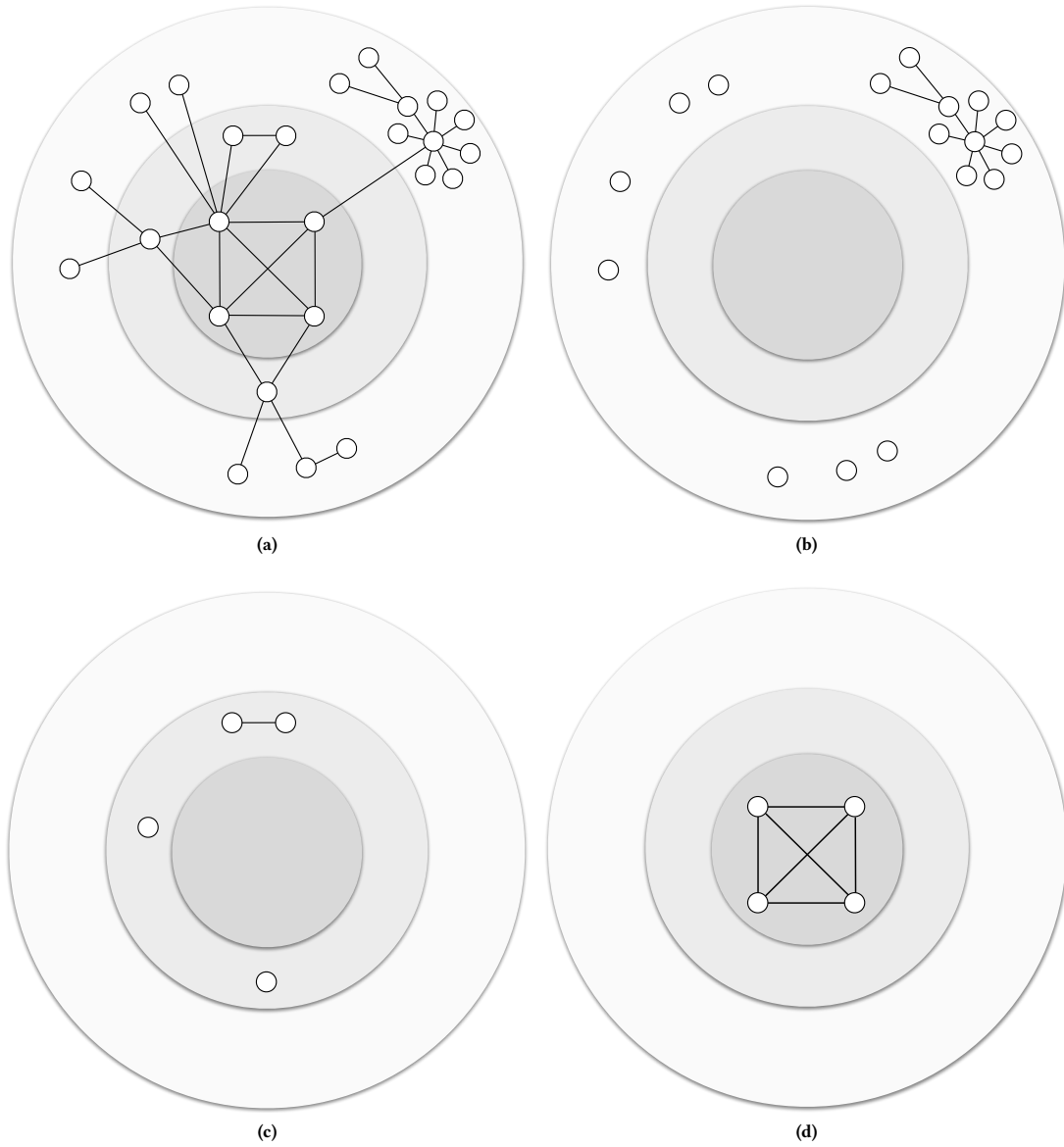
$$BC(v_i) = \sum_{st} n_{st}^i \quad (5)$$

However, there can be more than one shortest path between  $v_s$  and  $v_t$  and that will count for centrality measure more than once. Thus, if total number of shortest paths between  $v_s$  and  $v_t$  is  $g_{st}$ , the updated equation for finding betweenness centrality of node  $v_i$  will be,

$$BC(v_i) = \sum_{st} \frac{n_{st}^i}{g_{st}} \quad (6)$$

## 2.3 Susceptible-Infected-Recovered (SIR) Model

The SIR model is one of the simplest compartmental models in epidemiology. In an SIR model, every mode of a network can be at one of the following three states:



**Figure 2: A sample network with  $k_s$  value of the nodes**

- **S (Susceptible):** These denotes those people who have not been infected with the disease yet. However, they are not immune to it and therefore they are under threat of being infected with the disease in the future.
- **I (Infected):** These are people who have already been infected with the disease. Moreover, the infected people can transmit the disease to the susceptible neighbors with a probability of  $\beta$ .
- **R (Recovered):** These people have been recovered from the disease with a probability of  $\gamma$  and are immune now. Therefore, they are no longer under any threat to be infected with the disease in future.

The SIR model runs the simulation of disease diffusion based on the topology of the network and the probability parameters  $\beta$  and  $\gamma$ . The simulation stops when the network has no more infected nodes. This SIR model is capable of finding the most influential spreaders based on their spreadability.

### 3 RELATED WORK

An online social network (OSN) results from the use of a social network site (SNS) that allows the users to publish messages and connect to other users which result in creating social relationships. An OSN is formally represented by a graph, where nodes are users and edges are relationships that can be either directed or undirected.

In this graph the nodes play an important role to disseminate information. Finding the most influential spreader in an OSN has caught attention of the researchers. Recently more and more attentions have been paid to microscopically study the *spreadability* for each node. The knowledge of node *spreadability* is crucial for developing efficient methods to either decelerate spreading in the case of diseases, or speed up spreading in the case of information flow. Moreover, it can be helpful for identifying the initial spreader of certain disease or information.

Though the most connected nodes (hubs) and the nodes with high betweenness centrality are commonly believed to be the most influential spreaders in networks, the  $k$ -core (also called  $k$ -shell) method is found to perform better in identifying the best individual spreaders [7, 27]. Basically, the principle of the  $k$ -core decomposition is to assign a core index  $k_s$  to each node such that nodes with the lowest values are located at the periphery of the network while nodes with the highest values are located in the center of the network. We shall discuss the details about the algorithm shortly.

Cataldi et al. [8] proposed to use the well known PageRank algorithm [48] to assess the distribution of influence throughout the network. The *PageRank* value of a given node is proportional to the probability of visiting that node in a random walk of the social network, where the set of states of the random walk is the set of nodes. Both the methods only exploit the topology of the network, and ignore other important properties, such as nodes' features and the way they process information. Lü et al. [39] proposed the *LeaderRank* algorithm to identify influential spreaders in directed networks, which is a simple variant of *PageRank*, namely a so-called ground node connected with every other node by a bidirectional link is introduced into the original network, and then the standard random walk process is applied to dig out influential spreaders. Li et al. [33] further improved the *LeaderRank* by allowing nodes with more fans get more scores from the ground node. With almost the same converging speed (we have checked by simulations), this so-called *WeightedLeaderRank* performs better than *LeaderRank*.

Romero et al. [53] develop a graph-based approach similar to the well known HITS algorithm, IP (i.e. *Influence-Passivity*), that assigns a relative *influence* and a *passivity* score to every users based on the ratio at which they forward information. However, no individual can be a universal influencer, and influential members of the network tend to be influential only in one or some specific domains of knowledge. Therefore, Palet al. [49] developed a non-graph based, topic-sensitive method. To do so, they define a set of nodal and topical features for characterizing the network members. Using probabilistic clustering over this feature space, they rank nodes with a within-cluster ranking procedure to identify the most influential and authoritative people for a given topic. Weng et al. [57] also develop a topic-sensitive version of the *PageRank* algorithm dedicated to Twitter, *TwitterRank*. They presented the phenomenon of *homophily* in a community of Twitter. By making use of this phenomenon, a PageRank-like algorithm, called *TwitterRank*, is proposed to measure the topic-sensitive influence of the *twitterers*.

All these methods described are summarized in Table 1. We can see that none of the above approach is distributed. Most famous approach for finding influential spreader is the  $k$ -core decomposition which considers only the network topology. We propose a

distributed variant of the algorithm which also considers user info like no. of followers, no. of friends, whether the user is verified etc.

**Table 1: Summary of influential spreaders identification methods**

Algorithm	Network Topology	User Info	Topic Info	Distributed
$k$ -core decomposition	Y			
PageRank	Y			
Topic-sensitive PageRank	Y		Y	
IP		Y		
Topical Authorities		Y	Y	

## 4 OUR PROPOSED MODIFIED K CORE DECOMPOSITION METHOD

### 4.1 Distributed K Core Decomposition

Centralized algorithms for the  $k$ -core decomposition already exist [6]. Their algorithm is based on the recursive deletion of vertexes (and edges incident to them) of degree less than  $k$ . The algorithm makes use of *bin-sort*, and can run in  $O(\max(m, n))$ , which equals  $O(m)$  for connected networks. However a distributed variant is very much needed because of two possible scenarios: the graph could be so large to not fit into a single host, due to memory restrictions; or its description could be inherently distributed over a collection of hosts, making it inconvenient to move each portion to a central site. Montresor et al. [44] proposed a distributed algorithm for  $k$ -shell decomposition for very large graph. Their distributed algorithm is based on the property of locality of the  $k$ -core decomposition: due to the maximality of cores, the *coreness* of node  $u$  is the largest value  $k$  such that  $u$  has at least  $k$  neighbors that belong to a  $k$ -core or a larger core. The locality property tells that the information about the *coreness* of the neighbors of a node is sufficient to compute its own *coreness*. Based on this idea, the algorithm works as follows: each node produces an estimate of its own *coreness* and communicates it to its neighbors; at the same time, it receives estimates from its neighbors and use them to recompute its own estimate; in the case of a change, the new value is sent to the neighbors and the process goes on until convergence.

## 5 EVALUATION METRICS

In this section, we briefly present the metrics we use to evaluate the merit our proposed ranking technique against the already established ones. In the following two sections, we present the environment we use to run our experiments, present the datasets we use to cross check the and discuss the out

### 5.1 Modified Jaccard Similarity Coefficient

Jaccard similarity coefficient measures the similarity between two finite sets which is defined as the ratio of the intersection to the size of the union of the sample sets. Therefore, for two comparing sets  $A$  and  $B$ , the Jaccard similarity coefficient  $J(A, B)$  is,

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (7)$$

In this paper, we intend to measure the similarity of top  $n$  items from two different rankings. Let  $a$  and  $b$  are two comparing methods and  $A$  and  $B$  are the two generated rankings respectively,  $A_n$  and  $B_n$  are subsets of  $A$  and  $B$  respectively with top  $n$  elements, then we define modified Jaccard similarity coefficient  $J_m(A, B)@n$  as follows,

$$J_m(a, b)@n = \frac{|A_n \cap B_n|}{n} \quad (8)$$

We use this modified metric mainly to test the proportion of the common users in the two  $n$  sized sets of the most influential users obtained from the two comparing ranking algorithms. While comparing with a established method, the higher the overlap is, the more reliable the comparing ranking algorithm.

## 5.2 Rank Correlation Coefficient

In general, correlation analyses are bi-variate analyses that measure the strength of association between two variables and the direction of the relationship. In terms of the strength of relationship, the value of the correlation coefficient varies between +1 and -1. A value of  $\pm 1$  indicates that there exists a perfect degree of association between the comparing two variables. As the correlation coefficient value goes towards 0, this relationship between the two variables gets weaker. The  $\pm$  signs of the coefficient indicated the direction of the relationship; a + sign indicates a positive relationship while a - sign indicates a negative relationship.

In Section ??, we measure the correlation of the ranked list of users based on their spreadability generated by our modified K core decomposition against the rankings generated by other methods. We use two non-parametric rank correlations: Kendall's tau and Spearman's rank correlation coefficient.

**5.2.1 Kendall Tau Correlation Coefficient.** The Kendall tau rank correlation coefficient is used to test the similarities in the ordering of data when it is ranked by quantities. While other types of correlation coefficients use the observations as the basis of the correlation, Kendall's correlation coefficient uses pairs of observations and determines the strength of association based on the pattern of concordance and discordance between the pairs. Assume that  $L_1$  and  $L_2$  are the two rankings that are to be compared. Then Kendall analysis takes the following two properties into consideration:

- **Concordant:** Any pair of items  $(x_1, y_1)$  in  $L_1$  and  $(x_2, y_2)$  in  $L_2$  are considered as concordant if and only if they meet one of the following two conditions:
  - $(rank\_in\_L_1(x_1) > rank\_in\_L_2(x_2) \text{ and } rank\_in\_L_1(y_1) > rank\_in\_L_2(y_2))$
  - $(rank\_in\_L_2(x_2) > rank\_in\_L_1(x_1) \text{ and } rank\_in\_L_2(y_2) > rank\_in\_L_1(y_1))$
- **Discordant:** Any pair of items  $(x_1, y_1)$  in  $L_1$  and  $(x_2, y_2)$  in  $L_2$  are considered as discordant if and only if they meet one of the following two conditions:
  - $(rank\_in\_L_1(x_1) > rank\_in\_L_2(x_2) \text{ and } rank\_in\_L_1(y_1) < rank\_in\_L_2(y_2))$
  - $(rank\_in\_L_2(x_2) > rank\_in\_L_1(x_1) \text{ and } rank\_in\_L_2(y_2) < rank\_in\_L_1(y_1))$

Kendall Tau correlation co-efficient is denoted by  $\tau$ . If  $L_1$  and  $L_2$  are two different rankings with  $n$  similar elements,  $N(C)$  and

$N(D)$  represent the number of concordant and discordant pair respectively, then  $\tau$  can be calculated using the following equation:

$$\tau(L_1, L_2) = \frac{N(C) - N(D)}{\frac{1}{2}n(n-1)} \quad (9)$$

**5.2.2 Spearman's Rank Correlation Co-efficient.** Spearman's Rank correlation coefficient,  $R_s$  is a technique which can be used to summarize the strength and direction (negative or positive) of a relationship between two variables. The result will always be within 1 and minus 1. The closer  $R_s$  is to +1 or -1, the stronger the likely correlation. A perfect positive correlation is +1 and a perfect negative correlation is -1. Assume that  $L_1$  and  $L_2$  are two rankings of same  $n$  elements. For any element  $x$ , if the rankings of  $x$  in  $L_1$  and  $L_2$  are  $rank\_in\_L_1(x)$  and  $rank\_in\_L_2(x)$  respectively, then the distance of ranks,  $d = rank\_in\_L_1(x) - rank\_in\_L_2(x)$ . This value is squared to remove any negative values and When written in mathematical notation the Spearman Rank formula looks like this:

$$R_s = 1 - \frac{6 \sum d^2}{n^3 - n} \quad (10)$$

**5.2.3 Calculating Statistical Significance using Co-efficient Values.** Statistical significance is a measure of whether any research outcome are meaningful or not. In the field of hypothesis testing of statistics, the term *null hypothesis* is the default assumption that there is no association or relationship between two measured phenomena [18]. A result has statistical significance when it is very unlikely to have occurred given the null hypothesis [45]. To be specific, a significance level,  $\alpha$  is set for the experiment which denotes the probability of the experiment rejecting the null hypothesis, given that the null hypothesis were assumed to be true [13]; and the p-value of a result,  $p$  is the probability of obtaining a result at least as extreme, given that the null hypothesis were true. The result is statistically significant, by the standards of the study, when the condition  $p \leq \alpha$  holds [12, 15, 25, 29, 52]. for the evaluation of our ranking method, we set the significance level to 5%.

To measure the statistical significance of the result, we use the following formula to compute a z-value:

$$z = \frac{3 \times T \sqrt{n(n-1)}}{\sqrt{2(2N+5)}} \quad (11)$$

where  $T$  is the correlation co-efficient measured by the previously described techniques. Using the z-score, an area is found from a z-table. This area value is considered as the p-value of a result.

## 5.3 Normalized Discounted Cumulative Gain, NDCG

In this paper, we use Normalized Discounted Cumulative Gain, *NDCG* which is one of the widely used techniques to evaluate ranking systems. Let  $GT$  represents the weighted set of all users who generate the network. The weights are the relevance of the nodes to be selected as the most influential spreader on the network and the set  $GT$  is sorted according to this relevance value. We can refer to these relevance values and the ranking of the nodes in this set as our ground truth.

Now let  $X$  be the ranking of nodes to be the most significant spreaders identified by any comparing method. We define cumulative gain for first  $m$  rankings in  $X$ ,  $CG@m$  as:

$$CG@m = \sum_{i=1}^m rel_i \quad (12)$$

Where  $rel_i$  indicated the relevance value (from ground truth) of node at rank  $i$  in  $X$ .

Discounted cumulative gain ( $DCG$ ) penalizes each relevance value based on its rank in the results. Therefore, we define Discounted cumulative gain for first  $m$  rankings in  $X$  as:

$$DCG@m = \sum_{i=1}^m \frac{rel_i}{\log(i+1)} = \sum_{i=1}^m \frac{2^{rel_i} - 1}{\log(i+1)} \quad (13)$$

$IDCG$  is the  $DCG$  of the best possible results based on the ground truth. Therefore we define Ideal Discounted cumulative gain for first  $m$  rankings in  $GT$  as:

$$IDCG@m = \sum_{i=1}^m \frac{rel_i}{\log(i^{(I)} + 1)} = \sum_{i=1}^m \frac{2^{rel_i} - 1}{\log(i^{(I)} + 1)} \quad (14)$$

Where  $i^{(I)}$  indicates the Ideal rank of a node in  $GT$ .  $NDCG$  is obtained by dividing  $DCG$  by Ideal  $DCG$  ( $IDCG$ ), which normalizes the gain within  $[0, 1]$ . Therefore  $NDCG$  for first  $m$  rankings in  $X$  can be defined as,

$$NDCG@m = \frac{DCG@m}{IDCG@m} \quad (15)$$

In our evaluation section we use the metric  $NDCG@m(a, b)$  as the Normalized Discounted Cumulative Gain for first  $m$  elements of a ranking generated by method  $a$ , taking the ranking generated by method  $b$  as our ground truth.

## 5.4 Infection Rate on SIR Model

We use two different metrics related to infection rate on SIR model to evaluate our proposed method. These metrics were introduced by Ahajjam et al. [3] and are presented below:

**5.4.1 Infection Rate Function.** This metric is used in order to compare different method of finding most influential spreaders by simulating the network on SIR model with some of the top spreaders as initially affected ones. At any time  $t$ , the infection rate can be defined as,

$$IR(t) = \frac{N_I(t) + N_R(t)}{n} \quad (16)$$

Where  $IR(t)$  is infection rate at time  $t$ ,  $N_I(t)$  is number of infected nodes at time  $t$ ,  $N_R(t)$  is number of recovered nodes at time  $t$  and  $n$  is the total number of nodes.

**5.4.2 Final Infection Rate.** In order to investigate the fraction of nodes that is finally affected a metric  $IR(t_{max})$  is used. When the simulation reaches a steady state, if  $N_R(t_{max})$  is the number of finally recovered nodes, then this metric can be defined as,

$$IR(t_{max}) = \frac{N_R(t_{max})}{n} \quad (17)$$

## 6 EXPERIMENTAL SETUP

As we already have discussed, the global methods of finding most influential spreaders take large amount of time to generate the final result. Therefore, to evaluate the performance of our proposed modified K core decomposition method of finding most influential spreaders on twitter social network, we need to keep the size of the dataset considerably small. On the other hand, we run the experiments with on a distributed environment with large scaled network data which fails to run on a single computer because of memory overflow and/or take infeasible amount of time due to larger computational complexity. Consequently, we run our experiments on two different environments and they are described below:

### 6.1 Cross Validation on a Single Computer

Global metrics like closeness or betweenness centrality possess a very high computational complexity which makes them infeasible to apply on large datasets i.e. networks with large number of nodes and edges. Therefore, in order to cut off the time requirement during evaluating the performance of our proposed method against such global techniques with high complexity, we run all the comparing methods on network with smaller number of nodes and edges on computer with simple commodity hardware.

### 6.2 Large Network Analysis on Distributed Environment

A number of related works for the distributed and/or parallel processing of graph structures has been presented in the literature. One popular framework for massively parallelizing computational tasks is MapReduce [14], introduced by Google in 2004 for the parallel processing of large data-sets. While Map-Reduce can be used for processing graphs, its structure is not optimized for such tasks. This is the reason that led Google researchers to develop another framework, called Pregel [40] which is optimized for mining graphs data [22]. Apache Giraph [41, 42] is the open source version of Pregel built on top of hadoop. The main idea of Giraph is “think like a vertex”. The computation in Giraph consists of a sequence of iterations, called supersteps, during which the framework runs a user-defined function on each vertex. In this function, a node receives messages from neighbor nodes sent during the previous superstep, modifies its local state and sends messages to its neighbor nodes, to be received in the next superstep. Barrier synchronization is used, so that each superstep is separated from the next one. Individual nodes may leave the computation when they have reached the convergence to their final state. We use Apache Giraph for implementing distributed  $k$ -core decomposition algorithm as described in the previous subsection 4.1.

### 6.3 Methods Compared

We evaluate the merit of the ranking generated by our method against the ones generated by the following methods:

- Modified K Core Decomposition: Our method
- Degree Centrality
- Closeness Centrality
- Betweenness Centrality
- Eigenvalue Centrality

- HybridRank

We simulate the graph network of the datasets on the SIR model and as initially affected nodes, we use the top most influential spreaders generated by all the comparing methods. We use the implementation of SIR model from the python module EoN [1]. The input networks of the EoN module are NetworkX [21] graphs. Also we use this python module to find the centrality measures of the network.

## 6.4 Datasets Used

In order to test the performance of our proposed modified K core decomposition method of finding most influential spreaders on twitter social network, we mainly use \*\*\* real twitter datasets. Since we compare the ranking generated by our proposed method with that of the global techniques and the global techniques take too much time to generate the results, we consider multiple subsets of the main datasets with smaller number of nodes and edges. Below we briefly describe the datasets we use for evaluating our proposed method.

- We use the dataset from [30] which was collected by crawling the entire Twitter site for 6 months in 2009. This dataset contains 41.7 million of user profiles, 1.47 billion friend-follower relationships, 4,262 trending topics, and 106 million tweets. However, due to Twitter's new Terms of Services, this dataset has removed the tweet contents. Therefore, we only generate the friend-follower graph from this dataset. Since for cross validation with global techniques of finding most influential spreaders, we make two subgraphs from this dataset with smaller number of nodes and edges and we define them as follows.
  - **Kwak\_50K**: We generate a subgraph from the main dataset with 50,000 randomly selected nodes and edges connecting them in the main graph. Since at every run we select a different set of randomly selected graphs, the number of average incident edges on each node vary from 1.5 to 4. We refer this dataset to *Kwak\_50K* in the upcoming sections.
  - **Kwak\_100K**: Similar to the previous one, this dataset is another subgraph generated from the main one with randomly selected 100,000 nodes and their incident edges. We refer this dataset to *Kwak\_100K* in the upcoming sections.
- Another twitter dataset we use is collected by Kristina Lerman [31] in the year of 2010. It is a dataset containing 736,930 users and 36,743,448 links of social relationship among them. To make it feasible to run the global metrics on this dataset, we generate a smaller dataset out of this one also with randomly selected 100,000 nodes and their incident edges and we refer to this dataset as *Lerman\_100k* in our upcoming sections.
- **Twitter-Dynamic-Net**: Lou et al. [36] and Hopcroft et al. [24] collected this dataset for their research works. To collect this dataset, one of the known popular user on twitter was selected and then 10,000 of his/her followers were randomly collected. After that, these users were taken as seed users and a crawler was used to collect all followers of these users by traversing "following" relationships. The total number of users is 112,044. The crawler monitored the change of the

network structure among the 112,044 users during December, 2010 and finally obtained 443,399 dynamic friend-follower relationships between them. In our evaluation section, we refer to this dataset as *Lou\_Hopcroft*

One thing to be noted that, all the smaller datasets generated from the original ones are opted to be used for cross validation of the ranking generated by our proposed method. of against the established methods like

## 7 EXPERIMENTAL OUTCOME AND EVALUATION

First we compare the co-efficient metrics among our comparing methods to measure their similarity in determining most influential spreaders. After that, we simulate the network on SIR model with some of the top most influential spreads identified by each of the comparing methods. At the end of this section, we show the *NDCG* values of each of the rankings against the SIR generated one and two of the mostly used methods.

### 7.1 Measurement of Similarity Co-efficient among the Methods

For establishing a ground truth, we determine the real ranking of the nodes for every dataset based on their spreadability by simulating the SIR model on the network generated from the corresponding dataset. The SIR model is simulated for 100 times with  $\beta = .1$  and  $\gamma = 1$  and averaging the outcome, we determine the set of users ordered by their spreadability. Using the co-efficients defined in the section 5, this ranking is compared with the ranked list generated by each of the comparing methods. In addition we also compare the ranked list of modified k core decomposition with each of the other methods.

First, we measure modified Jaccard similarity co-efficient as defined in subsection 5.1,

### 7.2 Simulation on SIR Model

We simulate the SIR model to depict the disease diffusion based on the topology of our experimental datasets. For each method, the SIR model is simulated on the same dataset for five (5) times so that a steady result is obtained and the simulation was run for infection transition probability,  $\beta = 0.1$  and recovery probability,  $\gamma = 1$ . We observe the simulation upto a maximum time,  $t_{max} = 16$ . The simulations stop when a saturation in infection spreading is achieved and Figure 3 shows the infection rate achieved with the flow of time for each of the methods on each of the datasets. Every time we select top 10 most influential spreaders identified by the comparing methods and assign them as the initially affected nodes on the SIR model. We can clearly see that the infection rate while selecting the top 10 most influential spreaders obtained by our modified k core decomposition outperforms the that of the other methods. Even though for \*\*\*, the betweenness centrality provides a better result in a very slight fraction, our method can generate the accurate result in significantly lower computational complexity than the global methods.



Dataset	$\tau(SIR, MKC)$	$\tau(SIR, DC)$	$\tau(SIR, CC)$	$\tau(SIR, BC)$	$\tau(SIR, EC)$	$\tau(SIR, HR)$
Kwak_50K	0.87	0.76	0.78	0.85	0.71	0.87
Kwak_100K	0.91	0.70	0.70	0.83	0.77	0.89
Lerman_100k	0.81	0.71	0.80	0.82	0.73	0.82
Lou_Hopcoft	0.83	0.73	0.79	0.81	0.81	0.82

Dataset	$\tau(MKC, SIR)$	$\tau(MKC, DC)$	$\tau(MKC, CC)$	$\tau(MKC, BC)$	$\tau(MKC, EC)$	$\tau(MKC, HR)$
Kwak_50K	0.87	0.76	0.78	0.85	0.71	0.87
Kwak_100K	0.91	0.70	0.70	0.83	0.77	0.89
Lerman_100k	0.81	0.71	0.80	0.82	0.73	0.82
Lou_Hopcoft	0.83	0.73	0.79	0.81	0.81	0.82

Dataset	$J_m(SIR, MKC)@10$	$J_m(SIR, DC)@10$	$J_m(SIR, CC)@10$	$J_m(SIR, BC)@10$	$J_m(SIR, EC)@10$	$J_m(SIR, HR)@10$
Kwak_50K	0.87	0.76	0.78	0.85	0.71	0.87
Kwak_100K	0.91	0.70	0.70	0.83	0.77	0.89
Lerman_100k	0.81	0.71	0.80	0.82	0.73	0.82
Lou_Hopcoft	0.83	0.73	0.79	0.81	0.81	0.82

Dataset	$NDCG(SIR, MKC)@10$	$NDCG(SIR, DC)@10$	$NDCG(SIR, CC)@10$	$NDCG(SIR, BC)@10$	$NDCG(SIR, EC)@10$	$NDCG(SIR, HR)@10$
Kwak_50K	0.87	0.76	0.78	0.85	0.71	0.87
Kwak_100K	0.91	0.70	0.70	0.83	0.77	0.89
Lerman_100k	0.81	0.71	0.80	0.82	0.73	0.82
Lou_Hopcoft	0.83	0.73	0.79	0.81	0.81	0.82

## 8 FUTURE WORK

## 9 ACKNOWLEDGMENT

## 10 CONCLUSION

## REFERENCES

- [1] [n.d.]. Epidemics on Networks. <https://epidemicsonnetworks.readthedocs.io/en/latest/index.html>. [Online; accessed 07-December-2019].
- [2] Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. 2008. Finding high-quality content in social media. In *Proceedings of the 2008 international conference on web search and data mining*. ACM, 183–194.
- [3] Sara Ahajjam and Hassan Badir. 2018. Identification of influential spreaders in complex networks using HybridRank algorithm. *Scientific reports* 8, 1 (2018), 11932.
- [4] Roy M Anderson, Robert M May, and B Anderson. 1992. *Infectious diseases of humans: dynamics and control*. Vol. 28. Wiley Online Library.
- [5] Eytan Bakshy, Itamar Rosenn, Cameron Marlow, and Lada Adamic. 2012. The role of social networks in information diffusion. In *Proceedings of the 21st international conference on World Wide Web*. ACM, 519–528.
- [6] Vladimir Batagelj and Matjaž Zaveršnik. 2011. Fast algorithms for determining (generalized) core groups in social networks. *Advances in Data Analysis and Classification* 5, 2 (2011), 129–145.
- [7] Shai Carmi, Shlomo Havlin, Scott Kirkpatrick, Yuval Shavitt, and Eran Shir. 2007. A model of Internet topology using k-shell decomposition. *Proceedings of the National Academy of Sciences* 104, 27 (2007), 11150–11154.
- [8] Mario Cataldi, Luigi Di Caro, and Claudio Schifanella. 2010. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*. ACM, 4.
- [9] Wei Chen, Laks VS Lakshmanan, and Carlos Castillo. 2013. Information and influence propagation in social networks. *Synthesis Lectures on Data Management* 5, 4 (2013), 1–177.
- [10] Alex Cheng, Mark Evans, and Harshdeep Singh. 2009. Inside Twitter: An in-depth look inside the Twitter world. *Report of Sysomos, June, Toronto, Canada* (2009).
- [11] Reuven Cohen, Shlomo Havlin, and Daniel ben Avraham. 2003. Efficient Immunization Strategies for Computer Networks and Populations. *Phys. Rev. Lett.* 91 (Dec 2003), 247901. Issue 24. <https://doi.org/10.1103/PhysRevLett.91.247901>
- [12] Geoff Cumming. 2013. *Understanding the new statistics: Effect sizes, confidence intervals, and meta-analysis*. Routledge, 27–28 pages.
- [13] Peter Dalggaard. 2008. Introductory Statistics with R. Springer, Chapter Power and the computation of sample size, 155–262.
- [14] Jeffrey Dean and Sanjay Ghemawat. 2008. MapReduce: simplified data processing on large clusters. *Commun. ACM* 51, 1 (2008), 107–113.
- [15] Jay L Devore. 2011. *Probability and Statistics for Engineering and the Sciences* (8th ed.). Cengage learning. 300–344 pages.
- [16] Benjamin Doerr, Mahmoud Fouz, and Tobias Friedrich. 2012. Why rumors spread so quickly in social networks. *Commun. ACM* 55, 6 (2012), 70–75.
- [17] Sergey N Dorogovtsev, Alexander V Goltsev, and Jose Ferreira F Mendes. 2006. K-core organization of complex networks. *Physical review letters* 96, 4 (2006), 040601.
- [18] Norman R. Draper. 2011. The Cambridge Dictionary of Statistics, Fourth Edition by B. S. Everitt, A. Skrondal. *International Statistical Review* 79, 2 (2011), 273–274. [https://doi.org/10.1111/j.1751-5823.2011.00149\\_2.x](https://doi.org/10.1111/j.1751-5823.2011.00149_2.x) arXiv:[https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1751-5823.2011.00149\\_2.x](https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1751-5823.2011.00149_2.x)
- [19] S Fortunato. 2010. Community detection in graphs. *Phys. Rep.* (2010).
- [20] Linton C Freeman. 1977. A set of measures of centrality based on betweenness. *Sociometry* (1977), 35–41.
- [21] Aric Hagberg, Pieter Swart, and Dan Schult. [n.d.]. NetworkX: a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks. <https://networkx.github.io>. [Online; accessed 07-December-2019].
- [22] Minyang Han, Khuzaima Daudjee, Khaled Ammar, M Tamer Özsu, Xingfang Wang, and Tianqi Jin. 2014. An experimental comparison of pregel-like graph processing systems. *Proceedings of the VLDB Endowment* 7, 12 (2014), 1047–1058.
- [23] JAP Heesterbeek. 2000. *Mathematical epidemiology of infectious diseases: model building, analysis and interpretation*. Vol. 5. John Wiley & Sons.
- [24] John Hopcroft, Tiancheng Lou, and Jie Tang. 2011. Who will follow you back?: reciprocal relationship prediction. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 1137–1146.
- [25] Valen E Johnson. 2013. Revised standards for statistical evidence. *Proceedings of the National Academy of Sciences* 110, 48 (2013), 19313–19317.
- [26] Matt J Keeling and Pejman Rohani. 2008. *Modeling infectious diseases in humans and animals*. Princeton University Press.
- [27] Maksim Kitsak, Lazaros K Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Eugene Stanley, and Hernán A Makse. 2010. Identification of influential spreaders in complex networks. *Nature physics* 6, 11 (2010), 888–893.
- [28] Konstantin Klemm, M Ángeles Serrano, Víctor M Eguiluz, and Maxi San Miguel. 2012. A measure of individual role in collective dynamics. *Scientific reports* 2 (2012), 292.
- [29] Martin Krzywinski and Naomi Altman. 2013. Points of significance: Significance, P values and t-tests. , 1041–1042 pages.
- [30] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is Twitter, a social network or a news media?. In *WWW '10: Proceedings of the 19th international conference on World wide web*. ACM, New York, NY, USA, 591–600. <https://doi.org/10.1145/1772690.1772751>

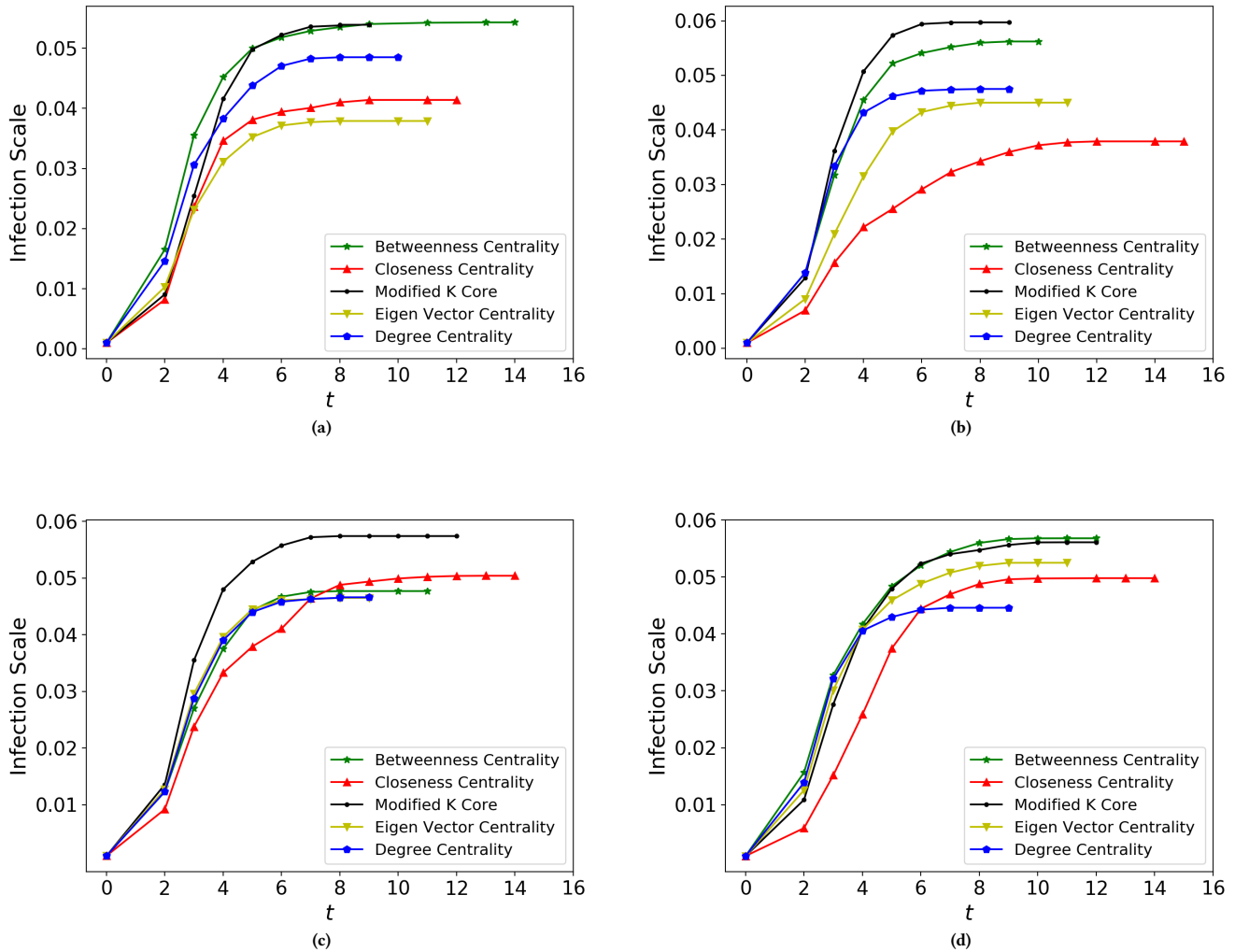


Figure 3: Simulation of SIR model on 4 datasets using each of the methods.

- [31] Kristina Lerman. 2010. Lerman Twitter 2010 Dataset. (2010). <http://www.isi.edu/~lerman/downloads/twitter/twitter2010.html>
- [32] Jure Leskovec, Lada A. Adamic, and Bernardo A. Huberman. 2007. The Dynamics of Viral Marketing. *ACM Trans. Web* 1, 1, Article 5 (May 2007). <https://doi.org/10.1145/1232722.1232727>
- [33] Qian Li, Tao Zhou, Linyuan Lü, and Duanbing Chen. 2014. Identifying influential spreaders by weighted leaderrank. *Physica A: Statistical Mechanics and its Applications* 404 (2014), 47–55.
- [34] Jian-Guo Liu, Jian-Hong Lin, Qiang Guo, and Tao Zhou. 2016. Locating influential nodes via dynamics-sensitive centrality. *Scientific reports* 6 (2016), 21380.
- [35] Ying Liu, Ming Tang, Tao Zhou, and Younghae Do. 2015. Improving the accuracy of the k-shell method by removing redundant links: From a perspective of spreading dynamics. *Scientific reports* 5 (2015), 13172.
- [36] Tiancheng Lou, Jie Tang, John Hopcroft, Zhanpeng Fang, and Xiaowen Ding. 2013. Learning to predict reciprocity and triadic closure in social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 7, 2 (2013), 5.
- [37] Linyuan Lü, Duanbing Chen, Xiao-Long Ren, Qian-Ming Zhang, Yi-Cheng Zhang, and Tao Zhou. 2016. Vital nodes identification in complex networks. *Physics Reports* 650 (2016), 1–63.
- [38] Linyuan Lü, Matúš Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, and Tao Zhou. 2012. Recommender systems. *Physics reports* 519, 1 (2012), 1–49.
- [39] Linyuan Lü, Yi-Cheng Zhang, Chi Ho Yeung, and Tao Zhou. 2011. Leaders in social networks, the delicious case. *PloS one* 6, 6 (2011), e21202.
- [40] Grzegorz Malewicz, Matthew H Austern, Aart JC Bik, James C Dehnert, Ilan Horn, Naty Leiser, and Grzegorz Czajkowski. 2010. Pregel: a system for large-scale graph processing. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*. ACM, 135–146.
- [41] C Martella. 2012. Apache Giraph: Distributed Graph Processing in the Cloud.
- [42] Claudio Martella, Roman Shaposhnik, Dionysios Logothetis, and Steve Harenberg. 2015. *Practical Graph Analytics with Apache Giraph*. Springer.
- [43] Sarah Milstein, Ben Lorica, Roger Magoulas, Gregor Hochmuth, Abdur Chowdhury, and Tim O'Reilly. 2008. *Twitter and the micro-messaging revolution: Communication, connections, and immediacy—140 characters at a time*. O'Reilly Media, Incorporated.
- [44] Alberto Montresor, Francesco De Pellegrini, and Daniele Miorandi. 2013. Distributed k-core decomposition. *IEEE Transactions on parallel and distributed systems* 24, 2 (2013), 288–300.
- [45] Jerome L Myers, Arnold D Well, and JR Lorch. 2010. Developing the fundamentals of hypothesis testing using the binomial distribution. *Research design and statistical analysis* (2010), 65–90.
- [46] Mark Newman. 2010. *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA.

- [47] Tore Opsahl, Filip Agneessens, and John Skvoretz. 2010. Node centrality in weighted networks: Generalizing degree and shortest paths. *Social networks* 32, 3 (2010), 245–251.
- [48] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank citation ranking: bringing order to the web. (1999).
- [49] Aditya Pal and Scott Counts. 2011. Identifying topical authorities in microblogs. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 45–54.
- [50] Romualdo Pastor-Satorras and Alessandro Vespignani. 2001. Epidemic spreading in scale-free networks. *Physical review letters* 86, 14 (2001), 3200.
- [51] Romualdo Pastor-Satorras and Alessandro Vespignani. 2002. Immunization of complex networks. *Physical review E* 65, 3 (2002), 036104.
- [52] Carol K Redmond and Theodore Colton. 2001. *Biostatistics in clinical trials*. John Wiley & Sons. 35–36 pages.
- [53] Daniel M Romero, Wojciech Galuba, Sitaram Asur, and Bernardo A Huberman. 2011. Influence and passivity in social media. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 18–33.
- [54] Gert Sabidussi. 1966. The centrality index of a graph. *Psychometrika* 31, 4 (1966), 581–603.
- [55] Stephen B Seidman. 1983. Network structure and minimum degree. *Social networks* 5, 3 (1983), 269–287.
- [56] Zhixiao Wang, Ya Zhao, Jingke Xi, and Changjiang Du. 2016. Fast ranking influential nodes in complex networks using a k-shell iteration factor. *Physica A: Statistical Mechanics and its Applications* 461 (2016), 171–181.
- [57] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. 2010. Twiterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 261–270.