2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) | 978-1-6654-5661-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ASONAM55673.2022.10068

# Identification of Key Actor Nodes: A Centrality Measure Ranking Aggregation Approach

Andreas Kosmatopoulos*, Kostas Loumponias†, Ourania Theodosiadou‡,
Theodora Tsikrika§, Stefanos Vrochidis¶ and Ioannis Kompatsiaris‖
*Information Technologies Institute*
*Centre for Research and Technology Hellas*
Thessaloniki, Greece
Email: *akosmato@iti.gr, †loumponias@iti.gr, ‡raniatheo@iti.gr, §theodora.tsikrika@iti.gr, ¶stefanos@iti.gr, ‖ikom@iti.gr

***Abstract*—The identification of key actors in complex networks has gathered significant interest by virtue of their importance in modern applications. Several of the existing methods employ standard centrality measures to achieve their goal and as a result, one of the main challenges is identifying key actor nodes with high relevance across all such measures. In this work, we propose a model based on the use of graph convolutional networks (GCNs) that retrieves the key actors in a network based on a centrality measure ranking aggregation scheme. We experimentally demonstrate the effectiveness of our solution compared to baseline and state-of-the-art approaches in terms of: i) accuracy, ii) performance compared to standard machine learning approaches, and iii) influence propagation capabilities.**

***Index Terms*—key actor, influence, social network, complex network, graph, neural networks**

## I. Introduction

A significant portion of modern applications employ complex networks to describe the elaborate relationships between their entities. Citation and collaboration networks, sensor networks and, more importantly, social networks can be intuitively modeled through graph data structures in which the different entities and their associations are represented through graph nodes and edges respectively. Over the past years, researchers have shown an increased interest in the identification of influential [30], [37] or critical nodes [18] in a graph due to their relevance in several subject areas, such as influence maximization [26], discovery of drug target candidates and proteins [9], network security and dismantling [28], [33], identification of influential spreaders [22], network immunization [24], rumour control [36] and others.

Several approaches to key actor[1] identification revolve around the use of classic node centrality measures, such as Degree Centrality [6], Betweenness Centrality [12], Closeness Centrality [13], Eigenvector Centrality [7] and PageRank Centrality [29]. While a single measure on its own cannot assess the overall importance of a node, it offers an intuitive way of ranking a node's influence inside the graph based on a structural property or information propagation capability [25].

It follows that, systems operating on complex networks under a particular "budget" of monetary or human resources (e.g. social workers, criminal investigators etc.) [3], [4] would benefit from focusing their analysis on a subset of nodes that exhibit high scores across all centrality measures. Thus, a need arises for the identification of key actor nodes that are "universally influential", i.e. nodes that are ranked higher among their peers with respect to all of the centrality measures.

In this work, we tackle this problem by proposing a GCN-based approach that combines local neighborhood centrality scores with the end goal of inferring an accurate aggregated ranking score for all nodes in the entirety of the graph. The model is trained around the expected ranking of the nodes according to the Borda Count voting rule [15] which is defined over the node's ranking in each of the centrality measures.

We experimentally showcase the accuracy of the proposed method compared to baseline methods and state-of-the-art influence maximization approaches. Furthermore, we demonstrate that when the percentage of key actors requested is small, our approach outperforms state-of-the-art methods in the SIR and LT models in terms of infected or activated nodes.

## II. Related Work

Research has been growing regarding the identification of influential nodes in graphs and subsequently several methods have been proposed. *VoteRank* constitutes an iterative method based on a voting procedure where the score of each node is affected by its neighbors [40]. Towards the direction of voting approaches, *VoteRank++* additionally defines the voting ability of each node as being proportional to its degree [23]. Moreover, an *improved K-shell* method has been proposed in order to identify key nodes based on the $k$-shell method considering also the impact of the neighbor nodes [34]. *EnRenew* algorithm uses the information entropy to identify influential nodes in a graph [14], while *RINF* is a re-ranking method in which the information spreading probability function is used to rank a certain node [38]. Finally, the *MCDE* method has been proposed that combines the core, degree, and entropy measures to rank a node [32], while *ECRM* takes into consideration the hierarchy of the nodes as well as their neighbors' [39].

Neural networks have been successful in many scientific fields for the last two decades. However, early variants of neural networks could not be implemented using (non-Euclidean) graph structure data [8]; this has led to the development of

[1]Throughout this work we use the terms "influential nodes" and "key actors" interchangeably.

IEEE/ACM ASONAM 2022, November 10-13, 2022
978-1-6654-5661-6/22/$31.00 © 2022 IEEE

2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

graph neural networks (GNN) [31]. One of the basic and well established variants of GNNs are the graph convolutional networks (GCNs) [17]. GCNs perform a similar operation as the plain convolutional neural networks (CNNs) [20]. However, a major difference between CNNs and GCNs is that CNNs are utilised on normal (Euclidean) structured data, while GCNs are the generalized version of CNNs, in which the numbers of edges vary and the nodes are unordered.

Finally, it is worth noting that Borda Count [15] voting schemes have been previously used in the context of machine learning methods and specifically as decision combination functions for multi-classifier systems [35].

## III. Proposed Framework

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph where $\mathcal{V}$ and $\mathcal{E}$ correspond to the vertex and edge set of $\mathcal{G}$ respectively. Given five centrality measures $\langle$degree, betweennness, closeness, eigenvector, pagerank$\rangle$ and their respective score rankings for each node

$$\mathcal{R} = \langle \mathcal{R}^{DC}, \mathcal{R}^{BC}, \mathcal{R}^{CC}, \mathcal{R}^{EC}, \mathcal{R}^{PC} \rangle \tag{1}$$

the Borda Count score of each node $v$ in $\mathcal{G}$ is

$$\mathcal{S}_v = \sum_{\mathcal{R}^i \in \mathcal{R}} \left( |\mathcal{V}| - \mathcal{R}^i_v \right) \tag{2}$$

where $\mathcal{R}^i_v$ corresponds to the rank of $v$ in score ranking $\mathcal{R}^i$.

The objective is to identify the set of nodes $\mathcal{D}$ with the $k$ highest Borda Counts:

$$\mathcal{D} = \underset{\substack{\mathcal{S}' \subset \mathcal{S} \\ |\mathcal{S}'| = k}}{\arg\max} \sum_{\mathcal{S}_v \in \mathcal{S}'} \mathcal{S}_v \tag{3}$$

where $\mathcal{S} = \{\mathcal{S}_j\}, j \in \mathcal{V}$ corresponds to the Borda Counts for all nodes in $\mathcal{G}$.

Additionally, we define the local neighborhood scores of a node $v$ to be the sequence

$$\mathcal{L}_{v,r} = \langle \mathcal{L}^{DC}_{v,r}, \mathcal{L}^{BC}_{v,r}, \mathcal{L}^{CC}_{v,r}, \mathcal{L}^{EC}_{v,r}, \mathcal{L}^{PC}_{v,r} \rangle \tag{4}$$

where $\mathcal{L}^i_{v,r}$ is the score of $v$ for centrality measure $i$ computed in its local neighborhood of radius $r$, i.e. a subgraph of $\mathcal{G}$ induced by all vertices with a distance up to $r$ from $v$.

Exhaustively evaluating the Borda Count for each node would be computationally prohibitive for large graphs. To tackle this issue, we propose a model (henceforth termed "ka-GCN", i.e., "key actor-GCN") that utilizes two GCN layers followed by two fully connected layers and provides an accurate Borda Count ranking approximation for a graph's nodes. To train the model, we annotate each node $v$ with its $\mathcal{L}_{v,r}$ scores for a hyperparameter $r$ and explicitly compute the $\mathcal{S}_v$ scores which serve as the target variable.

The model makes a prediction $\mathcal{S}^{(p)}_v$ for a node's Borda Count score, given its $\mathcal{L}_{v,r}$ scores. Since we are interested in the ranking of the $\mathcal{S}^{(p)}_v$ scores and not their absolute values, we make use of a ranking loss function to train the model [27]:

$$\text{Loss}(x, y) = \max\left(0, -y * \left(\mathcal{S}^{(p)}_v - \mathcal{S}_v\right) + \text{Margin}\right)$$
$$y = \begin{cases} 1 & \text{if } \mathcal{S}^{(p)}_v \text{ should be ranked higher than } \mathcal{S}_v \\ -1 & \text{if } \mathcal{S}_v \text{ should be ranked higher than } \mathcal{S}^{(p)}_v \end{cases} \tag{5}$$

The end result corresponds to an approximation of the Borda Count score for each node and can be used to find the key actor nodes in the graph.

## IV. Experimental Evaluation

In this section we showcase the effectiveness of "ka-GCN" under three different settings: i) The accuracy compared to indicative baseline centrality measures and state-of-the-art methods, ii) the performance compared to classic machine learning approaches, and iii) the spreading rate under certain conditions with regard to the SIR epidemiological model [1] and the LT diffusion model [16].

The model was pretrained on a collection of synthetic random graphs of varying topology. More specifically, we generated ten graphs for each of the following three topologies: Lancichinetti–Fortunato–Radicchi (LFR) [19], Erdős-Rényi (ER) [5], [10] and Barabási-Albert (BA) [2] and trained the model over 200 epochs with an Adam optimizer and a learning rate of $10^{-3}$. Specifically for the case of the LFR graphs we set the mixing parameter $\mu$ equal to $0.25$. The framework was developed using Pytorch and the DGL library. Graph manipulation and score computation was performed using graph-tool and NetworkX while synthetic graph generation was performed using Networkit. All experiments were run on an Intel Core i5-11600K CPU @ 3.90GHz machine with 12 cores, 16 GB RAM and an Nvidia GeForce RTX 3060 GPU.

### A. Dataset Description

Our experiments were conducted on three real world datasets retrieved from the SNAP [21] Dataset Collection: "ego-Facebook", "TVShows" and "ca-GrQc". "ego-Facebook" (2871 nodes and 62334 edges) contains social circles formed from users of Facebook, "TVShows" (3892 nodes and 17262 edges) contains a graph of verified Facebook pages and the mutual likes between them, and "ca-GrQc" (5242 nodes and 14496 edges) is a collaboration network of authors from the arXiv General Relativity and Quantum Cosmology category.

### B. Model Accuracy

In the first experiment we showcase the accuracy of our proposed model in identifying the top-$5\%$ key actors of a graph across the aforementioned centrality measures. We compare the performance of the proposed pre-trained model against baseline methods corresponding to centrality measures outputting the $5\%$ nodes with highest scores, as well as state-of-the-art methods for identifying influential nodes.

In this first experiment, it is important to note that these state-of-the-art methods were not originally designed to efficiently estimate scores across all centrality measures (akin

| | ego-Facebook | TVShows | ca-GrQc |
|---|---|---|---|
| Degree | 21.5% | 51.7% | 49.4% |
| Betweenness | 38.1% | 48.2% | 68.0% |
| Eigenvector | 5.5% | 34.3% | 41.8% |
| EnRenew | 15.2% | 26.6% | 20.1% |
| VoteRank [40] | 47.2% | 40.0% | 46.3% |
| VoteRank++ [23] | 13.1% | 22.5% | 26.6% |
| IKS [34] | 11.1% | 44.6% | 34.6% |
| ECRM [39] | 6.2% | 38.4% | 34.9% |
| ka-GCN | **59.0%** | **56.9%** | **70.0%** |

TABLE I: Model accuracy results

| | ego-Facebook | TVShows | ca-GrQc |
|---|---|---|---|
| LR | 65.1% | 54.2% | 46.0% |
| SVM | 62.7% | 64.4% | 44.4% |
| ka-GCN | **76.7%** | **66.1%** | **55.5%** |

TABLE II: Comparison with machine learning approaches

to Equation 3) and thus, they could be viewed as not being directly applicable to this problem. However, and in the absence of other applicable algorithms, we have decided to incorporate them as solid baseline choices.

We evaluate each method using the Accuracy metric [11]:

$$\text{Acc} = \frac{|\{\text{returned top-5\% nodes} \cap \text{actual top-5\% nodes}\}|}{\lceil |\mathcal{V}| \times 5\% \rceil} \tag{6}$$

Table I summarizes our results. We observe that in every dataset, "ka-GCN" achieves better performance than the rest of the baseline methods and retrieves the most accurate results with respect to the actual top-5% key actors (Equation 3). Furthermore, "Betweenness" and "VoteRank" offer a consistent alternative in all three datasets.

### C. Comparison with Machine Learning Approaches

In the second experiment, we examine the performance of "ka-GCN" compared to standard machine learning approaches. Specifically, we split each graph into a train and test subgraph consisting of 70% and 30% of the graph's starting nodes respectively and use the train dataset to build a logistic regression model and an SVM model. Finally, we train the "ka-GCN" model using the train dataset and compare the accuracy of the three approaches when tasked with retrieving the top-5% actors of the test dataset. Table II illustrates that "ka-GCN" remains effective even in the absence of pre-training.

### D. Model Propagation Capabilities

Further to the main purpose of this work, i.e. the identification of nodes with high values across all centrality measures (Equation 3), in this experiment we also study the propagation capabilities of the key actors identified by "ka-GCN" compared to those of the key actors identified by other state-of-the-art methods.

This analysis is performed using the SIR [1] and LT [16] models. In the SIR model a node exists in one of three states: Susceptible (S), Infected (I), and Recovered (R). Initially all nodes are set to (S) apart from a seed set of nodes that is set to (I). At each time moment, a node from (I) can infect a

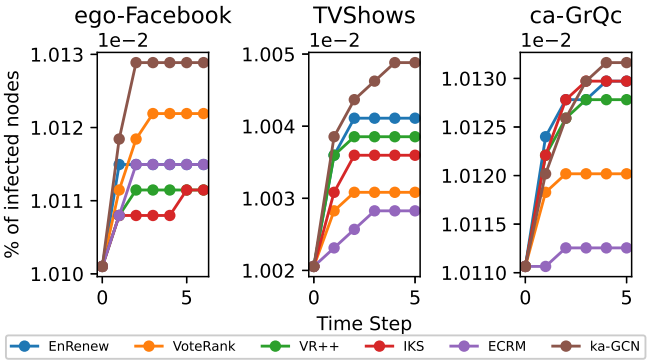


Fig. 1: Propagation rate under the SIR model

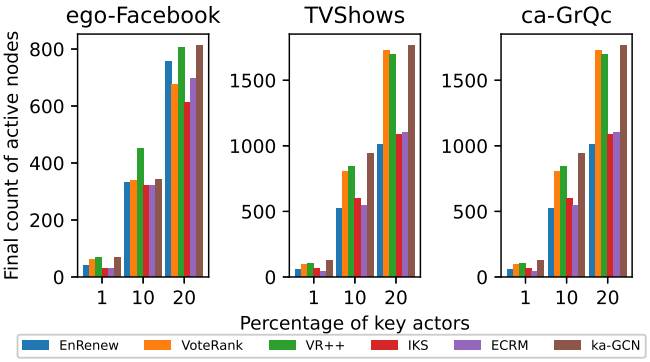


Fig. 2: Propagation rate under the LT model

neighbor node from (S) with probability $\mu$ turning their state into (I) as well. Additionally, at each time moment, a node from (I) turns its state to (R) with a probability of $\sigma$. This iterative process concludes when there are no significant node state changes between subsequent time moments.

We focus on the spreading capabilities of the top-1% key actors in each graph in an indicative setting where the infection and recovery rates are substantially low and high, respectively ($\mu = 10^{-3}$, $\sigma = 0.7$). Figure 1 demonstrates that under these conditions the key actors identified by "ka-GCN" have higher spreading capabilities than those of the other methods.

In the LT model we distinguish between active (A) and inactive (N) node states. At first, all nodes are in the (N) state, apart from a set of seed nodes which are in the (A) state. At each time point, an inactive node receives influence from its active neighbors. If the influence that the node received exceeds a predefined threshold $\tau = 0.75$, the node becomes active itself. Similarly to the SIR model, the iterative process stops when the model converges.

We study the propagation capabilities of each method when the percentage $k$ of key actors is 1%, 10% or 20% by measuring the total active node count when the model converges. Figure 2 shows that apart from the case of $k = 10\%$ in the "ego-Facebook" dataset, "ka-GCN" achieves better performance than the rest of the other methods in all cases.

## V. Conclusions

The identification of key actors (influential nodes) in a complex network is of high importance due to their relevance in many practical scenarios. Key actors typically exhibit high centrality scores and as a result identifying nodes with high values across all centrality measures would be advantageous for several applications. In this work, we proposed a GCN-based approach to identifying key actor nodes with high centrality measures by utilizing a ranking aggregation strategy and we experimentally demonstrated the effectiveness of the model compared to baseline and state-of-the-art methods.

## Acknowledgments

This project has received funding from the European Union's H2020 research and innovation programme as part of the CREST (GA No 833464), STARLIGHT (GA No 101021797) and INFINITY (GA No 883293) projects.

## References

[1] Roy M Anderson and Robert M May. *Infectious diseases of humans: dynamics and control*. Oxford university press, 1992.

[2] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[3] Kaustav Basu and Arunabha Sen. Epidemiological model independent misinformation source identification. In *Workshop Proceedings of the 15th International AAAI Conference on Web and Social Media*, 2021.

[4] Kaustav Basu and Arunabha Sen. Identifying individuals associated with organized criminal networks: a social network analysis. *Social Networks*, 64:42–54, 2021.

[5] Vladimir Batagelj and Ulrik Brandes. Efficient generation of large random networks. *Physical Review E*, 71(3):036113, 2005.

[6] Phillip Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of mathematical sociology*, 2(1):113–120, 1972.

[7] Phillip Bonacich. Power and centrality: A family of measures. *American journal of sociology*, 92(5):1170–1182, 1987.

[8] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[9] Peter Csermely, Tamás Korcsmáros, Huba JM Kiss, Gábor London, and Ruth Nussinov. Structure and dynamics of molecular networks: a novel paradigm of drug discovery: a comprehensive review. *Pharmacology & therapeutics*, 138(3):333–408, 2013.

[10] P. Erdös and A. Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290, 1959.

[11] Changjun Fan, Li Zeng, Yuhui Ding, Muhao Chen, Yizhou Sun, and Zhong Liu. Learning to identify high betweenness centrality nodes from scratch: A novel graph neural network approach. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 559–568, 2019.

[12] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[13] Linton C Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.

[14] Chungu Guo, Liangwei Yang, Xiao Chen, Duanbing Chen, Hui Gao, and Jing Ma. Influential nodes identification in complex networks via information entropy. *Entropy*, 22(2):242, 2020.

[15] Tin Kam Ho, Jonathan J. Hull, and Sargur N. Srihari. Decision combination in multiple classifier systems. *IEEE transactions on pattern analysis and machine intelligence*, 16(1):66–75, 1994.

[16] David Kempe, Jon Kleinberg, and Éva 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*, pages 137–146, 2003.

[17] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[18] Mohammed Lalou, Mohammed Amin Tahraoui, and Hamamache Kheddouci. The critical node detection problem in networks: A survey. *Computer Science Review*, 28:92–117, 2018.

[19] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical review E*, 78(4):046110, 2008.

[20] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.

[21] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[22] Hanwen Li, Qiuyan Shang, and Yong Deng. A generalized gravity model for influential spreaders identification in complex networks. *Chaos, Solitons & Fractals*, 143:110456, 2021.

[23] Panfeng Liu, Longjie Li, Shiyu Fang, and Yukai Yao. Identifying influential nodes in social networks: A voting approach. *Chaos, Solitons & Fractals*, 152:111309, 2021.

[24] Yang Liu, Xi Wang, and Jürgen Kurths. Framework of evolutionary algorithm for investigation of influential nodes in complex networks. *IEEE Transactions on Evolutionary Computation*, 23(6):1049–1063, 2019.

[25] Linyuan Lü, Duanbing Chen, Xiao-Long Ren, Qian-Ming Zhang, Yi-Cheng Zhang, and Tao Zhou. Vital nodes identification in complex networks. *Physics Reports*, 650:1–63, 2016.

[26] Lijia Ma, Zengyang Shao, Xiaocong Li, Qiuzhen Lin, Jianqiang Li, Victor CM Leung, and Asoke K Nandi. Influence maximization in complex networks by using evolutionary deep reinforcement learning. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2022.

[27] Sunil Kumar Maurya, Xin Liu, and Tsuyoshi Murata. Graph neural networks for fast node ranking approximation. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(5):1–32, 2021.

[28] Salomon Mugisha and Hai-Jun Zhou. Identifying optimal targets of network attack by belief propagation. *Physical Review E*, 94(1):012305, 2016.

[29] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[30] Sancheng Peng, Yongmei Zhou, Lihong Cao, Shui Yu, Jianwei Niu, and Weijia Jia. Influence analysis in social networks: A survey. *Journal of Network and Computer Applications*, 106:17–32, 2018.

[31] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[32] Amir Sheikhahmadi and Mohammad Ali Nematbakhsh. Identification of multi-spreader users in social networks for viral marketing. *Journal of Information Science*, 43(3):412–423, 2017.

[33] Sebastian Wandelt, Wei Lin, Xiaoqian Sun, and Massimiliano Zanin. From random failures to targeted attacks in network dismantling. *Reliability Engineering & System Safety*, 218:108146, 2022.

[34] Min Wang, Wanchun Li, Yuning Guo, Xiaoyan Peng, and Yingxiang Li. Identifying influential spreaders in complex networks based on improved k-shell method. *Physica A: Statistical Mechanics and its Applications*, 554:124229, 2020.

[35] Michał Woźniak, Manuel Grana, and Emilio Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16:3–17, 2014.

[36] Peng Wu and Li Pan. Scalable influence blocking maximization in social networks under competitive independent cascade models. *Computer Networks*, 123:38–50, 2017.

[37] En-Yu Yu, Yue-Ping Wang, Yan Fu, Duan-Bing Chen, and Mei Xie. Identifying critical nodes in complex networks via graph convolutional networks. *Knowledge-Based Systems*, 198:105893, 2020.

[38] Enyu Yu, Yan Fu, Qing Tang, Jun-Yan Zhao, and Duan-Bing Chen. A re-ranking algorithm for identifying influential nodes in complex networks. *IEEE Access*, 8:211281–211290, 2020.

[39] Ahmad Zareie, Amir Sheikhahmadi, Mahdi Jalili, and Mohammad Sajjad Khaksar Fasaei. Finding influential nodes in social networks based on neighborhood correlation coefficient. *Knowledge-based systems*, 194:105580, 2020.

[40] Jian-Xiong Zhang, Duan-Bing Chen, Qiang Dong, and Zhi-Dan Zhao. Identifying a set of influential spreaders in complex networks. *Scientific reports*, 6(1):1–10, 2016.