Influential Node Detection on Graph on Event Sequence

Zehao Lu[1(✉)], Shihan Wang[1(✉)], Xiao-Long Ren[2], Rodrigo Costas[3], and Tamara Metze[4]

[1] Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands
com3dian@outlook.com, s.wang2@uu.nl

[2] Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Chengdu, People's Republic of China

[3] Centre for Science and Technology Studies, Leiden University, Leiden, The Netherlands

[4] Technology, Policy and Management, Delft University of Technology, Delft, The Netherlands

Abstract. Numerous research efforts have centered on identifying the most influential players in networked social systems. This problem is immensely crucial in the research of complex networks. Most existing techniques either model social dynamics on static networks only and ignore the underlying time-serial nature or model the social interactions as temporal edges without considering the influential relationship between them. In this paper, we propose a novel perspective of modeling social interaction data as the graph on event sequence, as well as the Soft K-Shell algorithm that analyzes not only the network's local and global structural aspects, but also the underlying spreading dynamics. The extensive experiments validated the efficiency and feasibility of our method in various social networks from real world data. To the best of our knowledge, this work is the first of its kind.

Keywords: Influential Node Detection · Dynamics of Network · Non-epidemic Spreading

1 Introduction

Real-world networks exhibit high complexity as there are a large number and variety of nodes, interactions, or relationships. Therefore, modeling the spreading phenomenon (which could be either informative or physical) is difficult yet critical in a variety of fields such as infectious disease research [1], social media study [2], communication study [3]. The most intuitive way to find those opinion leaders in a network is to rank the nodes according to their influence. Numerous studies have been done to identify opinion leaders in various complex networks [4,5]. The majority of research on opinion leader detection in complex networks has assumed that the network is a static model, in which each node corresponds to an individual and the edges represent their long-term relationships.

H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2023, SCI 1143, pp. 147–158, 2024.
https://doi.org/10.1007/978-3-031-53472-0_13

However, many real-world social systems cannot be accurately depicted using static graphs due to their inability to account for temporal fluctuations in interactions and assume that the graph's structure remains unchanged [6]. The existing temporal network models mainly constructed network models by adding time-respecting edges on static models, therefore being able to model edge dissolving network phenomena.

Although the already existing network models are able to describe some network spreading mechanisms, these models overlook the relevant causal relationships within a sequence of events. Additionally, they fail to provide a quantification of influence. In some non-epidemic spreading situations (e.g. information spreading), people actively make decisions or react to received information rather than being passively affected [7]. In this case, the process of one person influencing another becomes a 'two-step' process: an actor first posts a message or commits a change, and another network member then reacts by doing the same (like posting or committing). The causal relationship here hinges on the chain of events, not just the network of individuals, as these active participants react to content or changes rather than just the sender [8]. These responsive actions or interactions can be summarized as '*events*' in a non-epidemic network. See the example in Fig. 1.

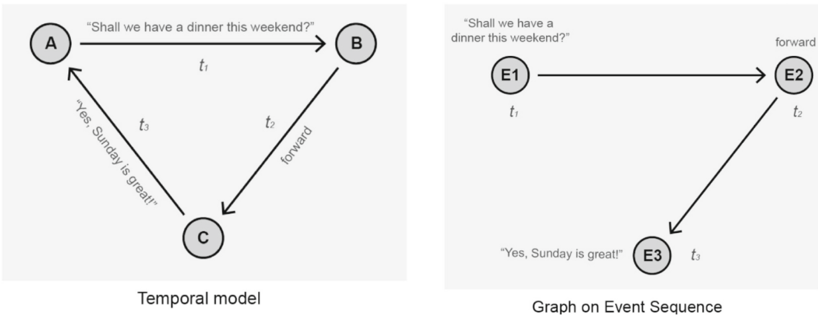


Fig. 1. A texting network example. The temporal model is not able to capture the affection relationship $E1 \rightarrow E2 \rightarrow E3$, as the adjacent edges do not necessarily have a direct affection relationship in a temporal graph.

In a texting network, the chain of messages is usually highly informative and has long-distance influences. Thus, sending the message can be conceptualized as an *event*. When A sends a message to B saying '*shall we (A,B,C) have a dinner this weekend?*' (first event), and B forward this to C afterward for confirmation (second event). Then, C texts A with '*Yes, Sunday is great!*' (third event). In this example, the third event is obviously a direct result of the first event, in other words, the first event influences the third event. However, the influence and chain-like relationships will not be captured in the current static or temporal network models (demonstrated in Fig. 1 left), as those models can only treat individuals as nodes and interactions as edges, while the relationship between edges (events) is usually ignored [6].

To fix the mentioned gap, this study offers a new perspective to model the graph on event sequence and to detect influential nodes in dynamic networks. In this model, we focus on social networks constructed with chains of informative events, and the influence of each social opinion is precisely measured by applying the Hawkes process model [9], which is a stochastic processes model that describes the progression of events through time, wherein the occurrence of prior events can influence the probability of future events. Furthermore, we propose a novel opinion leader mining method, the Soft K-shell, which functions on the graph on event sequence. The Soft K-shell algorithm applies the Hawkes process model on influence measurement and is able to use a variety of node properties (both topological and contextual) to find influential nodes. We conduct experiments on networks of different sizes and types to assess the Soft K-Shell's performance. The experimental results show that the proposed algorithm is feasible to perform better than the current benchmark algorithms.

2 Proposed Method

This section presents the proposed model (the graph on event sequence), and a related novel influential node detection algorithm (namely Soft K-shell).

2.1 Graph on Event Sequence

This research proposes a new type of social network graph model: the graph on event sequence. To make it clearer, the explanation of this term is given in this subsection. Unpack:

1: The **graph** is a directed structure (V, E) made of nodes V and edges E.
2: The **event sequence** is a sequence of interacting nodes $\{v_i\}$ in successive order. The temporal feature of the informatic flow is stored in the event sequence. The event sequence can be generated according to various situations. In the Fig. 2 example, to identify influential social media users we may define the top right event sequence, whereas the bottom right event sequence is more suitable for studying content of messages (e.g. influential scientific papers) in the social media community.
3: The term '**on**' here simply means that the graph itself is extracted from the event sequence.

Why 'Graph on Event Sequence'? One major disadvantage of the static graph model for studying social media is that they do not take the temporal features of information diffusion into account. That is to say, all interactions within the populations are considered equally in a static model, even if they are not. On the other hand, existing temporal models mainly consider network

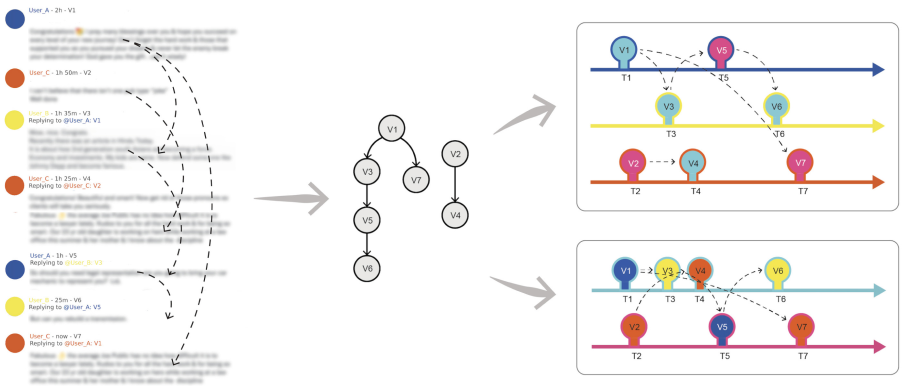


Fig. 2. A example of a graph on event sequence. The left plot illustrates an online social media screenshot, the colored circles (or 'bulbs') symbolize the network elements connected by the topological graph. Each 'bulb' (node) represents a post on social media. The extracted graph is shown in the middle, where each edge is an interaction between posts (e.g. reply or retweet). Interestingly, the same graph can be extracted from different event sequences (the right plots). For example, the top event sequences represent the posts from three users, where each user's posts are lined on the same arrow and the interaction between posts is represented by the dotted arrow. Here, the same contour color means the same users, while the filled color suggests the topic that the post is concerning. On the other hand, the bottom event sequences indicate the posts concerning two topics. In this event sequence, the contour color shows the topic and the filled color shows the publisher (e.g. Twitter user).

models only consisting of individuals while failing to construct a network for events. The graph on event sequence model provides an approach of combining the topological and temporal features for influential measurement to address the above issues, the graph on event sequence modeled the underlying dynamical process of events chain as a Hawkes process. Therefore, it is able to accurately describe the impact of each network event on its respondents. The graph on the event sequence model primarily serves as a content impact mining method, but it can also measure an individual's influence within a social network by summing the influences of events in a sequence, offering versatility for analyzing the impact of different content attributes, such as ranking influential journals based on their overall influence in a citation network.

2.2 Hawkes Process for Influence Measurement

The Hawkes process [9] is a mathematical model used in statistics and stochastic processes to describe the probability of occurrence of events $\{v_i\}$ over time using an intensity function. Hawkes processes is widely used in the area of influence measurement of non-epidemic spreading process [10–13]. The classic Hawkes Process is a counting process that models 'self-excited' events over a time period. In this research, a typical type of the multivariate Hawkes processes, the topological Haweks process model is applied to measure the influence of receiving information on reaction [14,15]. We propose to measure the impact of each network event on its respondents using the Hawkes process.

Definition 1. *A graph on event sequence is a direct graph $G = (V, E)$ that each node $v \in V$ is an element in an event sequence, a directed edge $e \in E$ represents the interaction relationship between the two nodes.*

Definition 2. *Given a threshold R, a graph $G = (V, E)$ and an influential function $I = I(v)$ where v is the node and $I(v)$ is the rank, a node $v \in V$ is called opinion leader if and only if it is in the top R nodes among the influentially ranking list that is sorted by $I(v)$.*

Definition 3. *The weight of each edge is defined as $W(e) = e^{-\beta(T(u)-T(v))}$ where u, v are the nodes connected by edge $e = v \rightarrow u$, $T(v)$ represents the timestamp when v occurred, and β is the scale parameter.*

Lemma 1. *Each connected component of the graph on event sequence is an acyclic graph.*

Proof. For any path in the graph on event sequence, the timestamp of successor nodes is later than their precursor's. Therefore if there is a cycle in any connected component, a time machine is invented.

Theorem 1. *Each connected component of the graph $G = (E, V)$ is a feed-forward network. For each node v in the graph, its direct successor nodes $S(v)$ is $\{u|u \in V, (v, u) \in E\}$, and the total influence of node v on its neighbor is,*

$$I(v) = \sum_{u \in S(v)} \alpha(u) e^{-\beta(T(u)-T(v))} \tag{1}$$

Proof. By lemma 1, we have that each connected component of the graph $G = (E, V)$ is directed and acyclic, thus, it is a feed-forward network. For each $u \in S(v)$, the influence of v on u is $\alpha(u)e^{-\beta T(u)-T(v)}$ By summing up those influences we have the above result. Note that the term α only depends on the property of node u, therefore it can be estimated by various machine learning or statistical algorithms.

Lemma 2. *For any two nodes v and u, the influence of v on u is $m \times \alpha(u)e^{-\beta(T(u)-T(v))}$ if $T(v) \leq T(u)$ and u, v both belong to a same connected component of G, otherwise the influence is 0. Here m is the number of distinct paths from v to u. If we do not consider any node properties then α is set to 1.*

Proof. Suppose u and v are below to same connected component of G, by Lemme 1, this connected component is an acyclic directed network. Thus there are m paths from v to u and none from u to v. Select one chain p, let $\{w_i\}$ denote the nodes on one of those chains from v to u as follows,

$$p : v \rightarrow w_1 \rightarrow w_2 \rightarrow ... \rightarrow w_n \rightarrow u \tag{2}$$

By the definition of intensity function and formula (1), there is,

$$I_u^p(v) = \alpha(u)e^{-\beta(T(u)-T(v))}$$

If there are m different chains from v to u, the summed influence should be,

$$I_u(v) = \sum_{p \in P(v,u)} I_u^p(v) = \sum_{p \in P(v,u)} \alpha(u)e^{-\beta(T(u)-T(v))} = m\alpha(u)e^{-\beta(T(u)-T(v))} \tag{3}$$

where $P(v, u)$ represents the set of paths from v to u. The influence of a node v on itself is $\alpha(v)e^{-\beta(T(v)-T(v))} = \alpha(v)$. If u, v do not belongs to a same connected component of G, $I_u(v)$ is naturally 0.

Theorem 2. *The overall influence of a node v in the graph can be obtained by directly computing its accumulated influence, which is defined as*

$$A(v) = \alpha(v) + \sum_{u \in S(v)} e^{-\beta(T(u)-T(v)} A(u) \tag{4}$$

where $S(v)$ is the set of direct successors of v. Note that m is not shown in formula (5) as there will only exist one direct path (an edge) for each neighbouring (u, v) pair.

Let $S'(v)$ denote the set of successors of v in G. By Lemma 2, there is

$$I_w^p(v) = \alpha(w)e^{-\beta(T(w)-T(v))} \quad \text{and,} \quad A(v) = \sum_{w \in S'(v)} \sum_{p \in \{P(v,w)\}} I_w^p(v)$$

where $p(v, w)$ denotes a direct path from v to w. Therefore,

$$\begin{aligned} A(v) &= \sum_{w \in S'(v)} \sum_{p \in \{P(v,w)\}} \alpha(w)e^{-\beta(T(w)-T(v))} \\ &= \alpha(v) + \sum_{w \neq v, w \in S'(v)} \sum_{p \in P(v,w)} \alpha(w)e^{-\beta(T(w)-T(v))} \end{aligned} \tag{5}$$

The summed form in (5) can also be written as $\sum_{w \neq v, w \in S'(v)} \sum_{p \in P(v,w)} \cdot = \sum_{p \in P(v,\cdot)} \cdot$ where $P(v, \cdot)$ represents all paths in G that start from v. Further, Note that any path p in G must include at least one node u which is a direct successor of v, one can split $P(v, \cdot)$ into $\bigcup_{u \in S(v)} P(u, \cdot)$, where u is the closest node to v on path p. And for u in $S(v)$,

$$e^{-\beta(T(w)-T(v))} = e^{-\beta(T(w)-T(u))}e^{-\beta(T(u)-T(v))} \tag{6}$$

Therefore, the summed form in (5) is equal to

$$\sum_w \sum_p \cdot = \sum_{p \in P(v,\cdot)} \alpha(w)e^{-\beta T(w)-T(v)} = \sum_{u \in S(v)} e^{-\beta(T(u)-T(v))} A(u) \tag{7}$$

By combining formula (5) and (7) there is,

$$A(v) = \alpha(v) + \sum_{u \in S(v)} e^{-\beta(T(u)-T(v))} A(u) \tag{8}$$

2.3 Soft K-Shell Algorithm

Accordingly, we propose a novel method, namely the Soft k-shell algorithm, that considers the topological Hawkes process of interaction. The proposed method addresses the task of detecting influential nodes by assessing the global influence of individual nodes in the graph using the Hawkes process and subsequently ranking them based on their overall influence[1]. The algorithm is executed as follows: first, if the node attribute is used, each node v's ranking is initially set to $\alpha(v)$, as it is proved by Theorem 2, the influence of a node v on itself is $\alpha(v)$. Otherwise, $\alpha(v) = 1$. It is advised that $\alpha(v)$ be set to a value between 0 and 1. Second, the global influence of node v whose direct successor u has a 0 degree ($\{u|u \in V, (v, u) \in E, degree(u) = 0\}$) is calculated recursively by adding its self influence (which is initialized as α) and its influence over u, which is $e^{-\beta(T(u)-T(v))}A(u)$. After performing this computation, node u will be permanently removed from the graph, and $A(u)$ is the final result of u's global influence. By repeatedly removing zero degree nodes us and updating the global influence of their predecessor vs, the graph G itself is also shrinking. As the graph is acyclic, there will always be new zero-out-degree nodes until all nodes are removed. We name this method the Soft K-shell algorithm. Two versions of the Soft K-Shell algorithm are considered here. If node-property is true then the property of nodes is used as $\alpha(v)$[2], otherwise $\alpha(v)$ is set to 1.

3 Experiments and Results

This study compares the performance of our method with four other methods for detecting opinion leaders in dynamic social networks. To evaluate the feasibility and generalizability of our method, six real world data sets of various types are used. Four of them are a collection of long-term Dutch tweets containing Coronavirus tags from February 2020 to January 2021 [16], which are split into four different data sets (NCF(reply), NCF(quote), NCF(retweet), NCF(together)) depending on their interaction type. The fifth data set (NCJ) contains short-term Dutch tweets related to the COVID-19 pandemic (three hours around a pandemic press conference on 14th January 2022) [17], while the sixth data set (DBLP V1) is the first version of the DBLP dataset (citation network) [18]. The specifications of the data sets utilized in the studies and the source code can be found in the paper's repository.

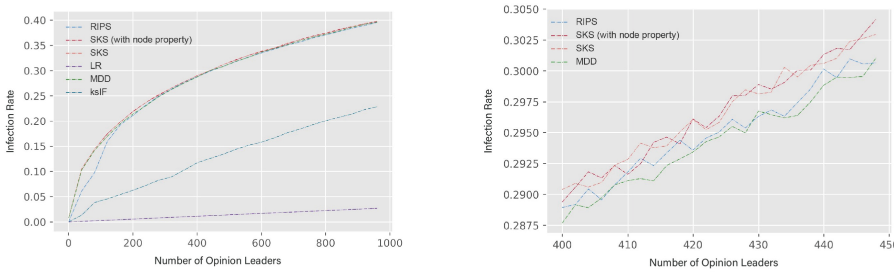
3.1 SIR Simulation Results

In this study, we compare the experimental results of our method with those of four other state-of-the-art algorithms, utilizing the Susceptible-Infected-

[1] A pseudocode of the proposed Soft K-shell algorithm could be found in this paper's repository, https://github.com/com3dian/SoftKShell.

[2] For the soft k-shell model, the parameter of node properties α is user-defined. In our conceptual framework, this value is assumed to be calculated using other machine learning techniques and given as the prior knowledge. Consequently, we do not delve into the methodologies for obtaining this parameter in the paper.

Removed (SIR) [19] simulation infection rate and computational complexity. The SIR model is a widely used framework for simulating the diffusion of information within networks. In scenarios where opinion leaders are designated as the initial set of infected nodes, upon achieving convergence, this model yields an infection rate that serves as a quantitative measure of the influence exerted by these initial opinion leaders throughout the entire information diffusion process. The transmission rate τ of each edge is 0.98 and the recovery rate γ of each node is 0.02. In every simulation, the top 5% ranked nodes were selected and infected initially. The simulations were conducted for 50 rounds. During our experiments, we used the number of followers as α for the NCF/NCJ datasets and citation numbers as α for the DBLP dataset.



(a) The SIR results from a larger scale.

(b) The SIR results from a smaller scale.

Fig. 3. Following the SIR model, the infection rates of various approaches on the NCF(together) network are presented (with different zoom scales). The x-axis shows the number of the initial opinion leaders (seed set), while the y-axis shows the final proportion of infection (in percentage). All comparable methods use the same size of opinion leaders' set as the seed set of SIR. The plot shows that the proposed method, Soft K-Shell, outperforms all other methods. The final SIR infection rate does not significantly differ between the two versions of the Soft K-Shell (with and without node properties). MMD and RIPS also perform well.

The results of Soft K-Shell (SKS) and several other state-of-the-art methods in the literature were used to validate the proposed model and algorithm. LeaderRank algorithm (LR) [20], mixed degree decomposition (MDD) [21], k-shell iteration factor (ksIF) [22], and Randomized Influence Paths Selection (RIPS) [23] are selected as the baseline. The first three algorithms are widely used baselines in opinion leader mining tasks, while the RIPS is the state-of-art algorithm according to its results [23]. The parameters used in the following experiment are suggested by its authors.

In particular, four versions of the RIPS algorithm are used for comparison. Two versions consider the Hawkes intensity $e^{-\beta(T(u)-T(v)}$ as edge weight while the other two use equal weight for all edges. Additionally, we consider two versions of the Soft K-Shell algorithm in the experiments (with node property or not). Figures 3a and 4 depict the SIR model infection rates on the NCF(together)

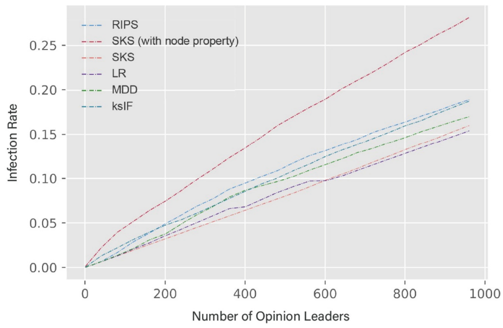


Fig. 4. In the NCJ dataset, the Soft K-Shell algorithm with node property achieves the highest infection rate, while the K-Shell iterative factor algorithm also performs well. Notably, the performance between the two versions of Soft K-Shell differs a lot, indicating the significant impact of node property in enhancing algorithm performance. This enhancement can be attributed to the characteristics of the NCJ dataset, which mainly consists of short-term Twitter interactions. In short-term information diffusion, users with a larger follower count exert a more significant influence. This phenomenon enhances the advantage of the Soft K-Shell algorithm with respect to its feasibility in combining node properties.

data set and the NCJ data set respectively. To better show the subtle difference in Fig. 3a, we also present plots of infection rates from the smaller scale in Fig. 3b.

Table 1 presents the highest infection rates obtained through SIR simulations across all six datasets, considering various algorithm settings. Our proposed algorithm consistently achieved the highest infection rate across datasets of varying sizes and social interactions. This demonstrates the feasibility and generalizability of our opinion leader mining method.

Table 1. The infection rate after SIR model reached the convergence of different methods results. Top 5% nodes found by each algorithm are used in the SIR model. The highest score on each data set is bolded. The Soft K-shell algorithm outperforms every other model.

Datasets	LR	MMD	Ksif	RIPS	**SKS**	**SKS** with node property
NCF(reply)	0.048	0.087	0.102	0.092	**0.133**	**0.133**
NCF(quote)	0.155	0.240	0.199	0.212	**0.250**	**0.250**
NCF(retweet)	0.050	0.481	0.373	0.469	**0.490**	0.488
NCF(together)	0.050	0.481	0.373	0.469	0.489	**0.490**
NCJ	0.056	0.067	0.067	0.076	0.050	**0.108**
DBLP V1	0.100	0.164	0.140	0.181	0.198	**0.210**

Table 2. Computational complexity of different methods

Algorithms	Complexity
K-Shell	$O(\|V\|^2)$
MDD	$O(\|V\|^2)$
ksIF	$O(\|V\|^2 + \|E\|)$
RIPS	$O(\|V\|log(\|V\|) + 2\|E\|)$
Soft K-Shell	$O(\|V\|^2)$

3.2 Computational Complexity Results

In this paper, we also assess the proposed method based on the computational complexity. The time complexity of five algorithms is listed in Table 2. The Soft K-shell algorithm has the same time complexity as the K-shell algorithm. In terms of the number of nodes $|V|$, RIPS appears to be the most effective algorithm because its highest ordered component is $|V|log(|V|)$, whereas other algorithms have the term $|V|^2$. However, in many real world networks, the amount of edges $|E|$ has the same order as $|V|^2$. Additionally, because the RIPS technique uses a Monte-Carlo-based methodology, the constant coefficients in the asymptotic complexity term are significantly larger than those in the other four methods.

3.3 Soft Shell Decomposition

Besides the quantitative results, we had another interesting discovery even though the Soft K-Shell algorithm does not compute a 'hard' decomposition of the network, its computed node ranking follows a multimodal distribution. As demonstrated in Fig. 5, the scatter plot of the NCF(together) data set has four 'shells' which together can be described as a soft shell decomposition. This is also the inspiration for naming this algorithm. The four 'shells' from the inside out each represent one type of posts in the NCF(together) dataset. The most centered shell (purple) represents the most influential tweets; while the second shell (deep blue) represents the posts that have a small range impact in the 'local' social network; the third shell (green) represents the most ordinary posts; the fourth shell (yellow) represents the 'dead' posts that are rarely noticed by any other people. It can be concluded that the Soft K-shell ranking result is also able to reveal the soft shell nature of a network. The above soft shell decomposition result not only gives the ranking of posts regarding their influence, but also gives a distribution of their influence. With the help of that distribution, a more specific community opinion study on the purple shell can be carried out, since the purple shell is highly representative of the community and is much smaller compared to the entire network. As a result, the soft shell decomposition can immensely benefit the analysis of social media platforms' processes for forming opinions based on their content.

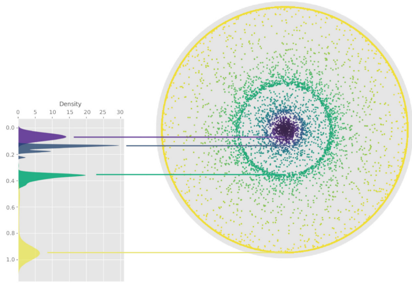


Fig. 5. The shell decomposition or the coreness decomposition of the NCF(together) data set. Each data point is a rank of a node in NCF(together)(which representing a post on Twitter), which has been scaled by exponential function $e^{1-rank(v)}$, where $rank(v)$ is ranging from 1 to $+\infty$. The coreness trait is also seen in the result of Soft K-Shell algorithm, despite the fact that it does not use the original K-Shell technique.

4 Conclusion

With the increasing popularity of social networks, resolving essential problems about these networks, such as opinion leader detection, has gained a lot of interest. However, most of the existing static and temporal methods fail to explain the underlying dynamic of non-epidemic spreading process in the networks. Therefore, a new method that considers the Hawkes process of information flow to combine the topological and temporal features for influential measurement, is proposed in this paper. The proposed model outperforms the state-of-the-art model by a significant margin while keeping a competitive time complexity. In future work, more theoretical study shall be accomplished on finding the scale parameter β and the node's property α. Also, we plan to investigate more into using the time serial structure of the Hawkes process to forecast social network dynamics and its impact on public opinion.

References

1. Goh, K.-I., Cusick, M.E., Valle, D., Childs, B., Vidal, M., Barabási, A.-L.: The human disease network. Proc. Natl. Acad. Sci. USA **104**, 8685–8690 (2007)
2. Vespignani, A.: Modelling dynamical processes in complex socio-technical systems. Nat. Phys. **8**, 32–39 (2012)
3. Garton, L., Haythornthwaite, C., Wellman, B.: Studying online social networks. J. Comput.-Mediat. Commun. **3**, JCMC313 (1997)
4. Bamakan, S.M.H., Nurgaliev, I., Qu, Q.: Opinion leader detection: a methodological review. Expert Syst. Appl. **115**, 200–222 (2019)
5. Lü, L., Chen, D., Ren, X.-L., Zhang, Q.-M., Zhang, Y.-C., Zhou, T.: Vital nodes identification in complex networks. Phys. Rep. **650**, 1–63 (2016)
6. Holme, P., Saramäki, J.: Temporal networks. Phys. Rep. **519**(3), 97–125 (2012). Temporal Networks

7. Zheng, M., Lü, L., Zhao, M.: Spreading in online social networks: the role of social reinforcement. Phys. Rev. E **88**, 012818 (2013)
8. Inwagen, P.V.: An Essay on Free Will. Oxford University Press, New York (1983)
9. Hawkes, A.G.: Spectra of some self-exciting and mutually exciting point processes. Biometrika **58**(1), 83–90 (1971)
10. Kobayashi, R., Lambiotte, R.: TiDeH: Time-dependent Hawkes process for predicting retweet dynamics. Proc. Int. AAAI Conf. Weblogs Soc. Media **10**, 191–200 (2021)
11. Zadeh, A.H., Sharda, R.: Hawkes point processes for social media analytics. In: Iyer, L.S., Power, D.J. (eds.) Reshaping Society through Analytics, Collaboration, and Decision Support. AIS, vol. 18, pp. 51–66. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-11575-7_5
12. Alvari, H., Shakarian, P.: Hawkes process for understanding the influence of pathogenic social media accounts. In: 2019 2nd International Conference on Data Intelligence and Security (ICDIS), pp. 36–42, IEEE (2019)
13. Filimonov, V., Sornette, D.: Quantifying reflexivity in financial markets: toward a prediction of flash crashes. Phys. Rev. E **85**, 056108 (2012)
14. Cai, R., Wu, S., Qiao, J., Hao, Z., Zhang, K., Zhang, X.: THP: topological hawkes processes for learning granger causality on event sequences. arXiv preprint arXiv:2105.10884 (2021)
15. Embrechts, P., Liniger, T., Lin, L.: Multivariate hawkes processes: an application to financial data. J. Appl. Probab. **48**(A), 367–378 (2011)
16. Colavizza, G., Costas, R., Traag, V.A., van Eck, N.J., van Leeuwen, T., Waltman, L.: A scientometric overview of cord-19. PLoS ONE **16**(1), e0244839 (2021)
17. Wang, S., Schraagen, M., Sang, E.T.K., Dastani, M.: Dutch general public reaction on governmental COVID-19 measures and announcements in twitter data. arXiv preprint arXiv:2006.07283 (2020)
18. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. In: AAAI (2015)
19. Ross, R.: An application of the theory of probabilities to the study of a priori pathometry.-Part I. Proc. Roy. Soc. Lond. A **92**(638), 204–230 (1916)
20. Lü, L., Zhang, Y.-C., Yeung, C.H., Zhou, T.: Leaders in social networks, the delicious case. PLoS ONE **6**(6), e21202 (2011)
21. Zeng, A., Zhang, C.-J.: Ranking spreaders by decomposing complex networks. Phys. Lett. A **377**(14), 1031–1035 (2013)
22. Wang, Z., Zhao, Y., Xi, J., Du, C.: Fast ranking influential nodes in complex networks using a K-shell iteration factor. Phys. A: Stat. Mech. Appl. **461**, 171–181 (2016)
23. Rezaei, A.A., Jalili, M., Khayyam, H.: Influential node ranking in complex networks using a randomized dynamics-sensitive approach. arXiv preprint arXiv:2112.02927 (2021)