DepthRank: Exploiting Temporality to Uncover Important Network Nodes

Nikolaos Bastas *, Theodoros Semertzidis, and Petros Daras
Centre for Research and Technology Hellas, Thessaloniki, Greece
{nimpasta,theosem, daras}@iti.gr

Abstract. Identifying important network nodes is very crucial for a variety of applications, such as the spread of an idea or an innovation. The majority of the publications so far assume that the interactions between nodes are static. However, this approach neglects that real-world phenomena evolve in time. Thus, there is a need for tools and techniques which account for evolution over time. Towards this direction, we present a novel graph-based method, named DepthRank (DR) that incorporates the temporal characteristics of the underlying datasets. We compare our approach against two baseline methods and find that it efficiently recovers important nodes on three real world datasets, as indicated by the numerical simulations. Moreover, we perform our analysis on a modified version of the DBLP dataset and verify its correctness using ground truth data.

Keywords: Influence Detection, Network Analysis, Temporal Awareness

1 Introduction

In everyday life, humans communicate with each other in various ways: face-to-face [20], through mobile devices [18], facebook posts [24] or tweets [6], to name a few. Graph theory is a valuable tool for investigating these relations because it provides analytic capabilities, meaningful representations and easy visualization of the results [15]. In this setting, humans are considered as nodes and their communication as links. In various applications, it is important to find those nodes that play a dominant role in the interconnection of the graph and the exchange of information: examples are the stability of power grid supply [17], the diffusion of ideas or innovations [3], the delay of an epidemic outbreak [5]. We are interested in those nodes that maximize the spread of information, known as influence maximization [7]. There is a vast amount of research in this direction using either network topology, network functionality or both, on static representations. For a recent review, the reader is refered to [11] and references therein.

The static representations neglect the evolution over time of the real world networks which may seriously affect the outcome. For example, in Fig. 1, users

* Corresponding author
A and W appear at different time steps (denoted by the separate planes in Fig. 1 (a)). They exhibit the same connectivity pattern (i.e. A(W) $\rightarrow$ B $\rightarrow$ C and A(W) $\rightarrow$ B $\rightarrow$ D). This is captured by the static representation in Fig. 1(b). According to the latter, the two users are symmetric with respect to user B and there is no difference on how their influence is passed through it. However, A appears earlier in this sequence (Fig.1(a)) and thus its “influence” should have decayed compared to the “influence” exerted by W on the same set of subsequent users. This cannot be captured by the “static” representation.

Fig. 1. (a) An example of a temporal network and (b) its static representation. In (a) A and W appear at different time steps ($t_1$ and $t_3$, respectively), having the same connectivity pattern. Thus, it is expected that their “influence” on the rest of the nodes should not be the same due to aging effects. In (b) the connectivity pattern is retained but the time ordering is neglected; this representation indicates an equal behavior for both A and W, which biases the outcome.

In the last few years, there has been an increasing interest for tools and techniques that preserve the temporal properties of the graphs. In [1], the authors formulate stochastic evolutionary equations for information spreading on a time varying network. A similar approach was used in the field of epidemic spreading [16]. In [25], the authors exploit the in-degree of the nodes on an evolving network, to determine the set of users that maximize the spread of influence under the Independent Cascade (IC) model [7]. In [13], the authors divide the dataset into two parts, using the first part for the identification of possible seed sets, and the second part for the evaluation of the spreading ability of the seed sets under a temporal Linear Threshold (LT) model [7]. In [19], the influence maximizing set is evaluated by preserving the dynamic nature of the underlying network and developing an efficient update scheme. In [10], a ranking method is proposed based on communicability [4] combined with an aging function, which is called “dynamic centrality”. A similar approach is used on opportunistic mobile social networks [2]. In [12], the authors define the *temporal delta-centrality* to measure the importance of each node as time passes.
In this paper, we propose a novel method which calculates the ability of each node to transmit information by sequentially processing fixed time windows and incorporating an aging factor. The outcome of the proposed method is a ranking of the network nodes which indicates their strength in influencing other nodes within the evolving graph.

The rest of the paper is organized as follows: In Section 2, we briefly outline our approach which is fully described in Appendix A. In Section 3 we present in short the baseline methods and the evaluation process with the details left for Appendices B and C. In Section 4 we illustrate the obtained results, while in Section 5, we provide a summary and propose future directions.

## 2 Method

Here, we briefly present our proposed method (for more see Appendix A). We assume that the sequence of interactions of the form \((u, v, t)\) is mapped onto a set of directed unweighted graphs \(G_i, i = 1, ..., M\) - called layers - by aggregating them within user-defined time windows. Each layer \(G_i\) corresponds to a time step \(t_i\). DepthRank is a two level procedure, incorporating (a) the calculation of the "influence score" update for each node in consecutive time steps and (b) the application of a monotonically decreasing function to account for the influence decay as time passes. With respect to (b), we introduce three variants of DepthRank: with no forgetting mechanism (DR), with forgetting mechanism (DR - F) and with forgetting mechanism with normalization (DR - NF). The "influence score" \(S\) of a node \(s\) taking into account all the interactions in the dataset is given by the following set of equations in the case of DR-F:

\[
\Delta H(s, G_i) = \sum_{w \in G_i+d} H(w, G_i+d|s, G_i) \quad (1)
\]

\[
S(s, G_i) = \Delta H(s, G_i) + S(s, G_j) \cdot e^{-(t_i-t_j)} \quad (2)
\]

\[
S(s) = S(s, G_j) \cdot e^{-(T-t_j)} \quad (3)
\]

In the previous equations, \(H(w, G_i+d|s, G_i)\) is the product of incoming and outgoing links to node \(w\) in layer \(G_i+d\), given that there is a path from node \(s\) at layer \(G_i\) (see Appendix A). \(\Delta H(s, G_i)\) refers to the update of the "influence score" of node \(s\) rooted in layer \(G_i\) and \(S(s, G_j)\) is the score acquired after the most recent update of \(S\) at time step \(t_j\), corresponding to layer \(G_j\). Note that \(t_j < t_i\). The final "influence score" for node \(s\) is calculated in Eq. (3). \(T\) is the maximum time step in the aggregated data. The rest of the cases (DR and DR-NF), along with the notations and algorithmic implementation are described in Appendix A.

## 3 Experimental setup

Our experiments were performed using a Ubuntu 14.04 LTS desktop PC with Intel(R) Core(TM) i5-2500K CPU at 3.30 GHz and 16 GB RAM. For comparison,
we have employed two popular and widely used methods, namely $k$-shell (kS) [9] and Collective Influence (CI) [14] (see Appendix B). To assess the spreading ability of the important network nodes uncovered by each method, we apply a temporal implementation of the SIR model, called Temporal Gillespie Algorithm [23], using as an indicator the maximum number of infected nodes within the time limits of the dataset $I_{max} = \max_{1 \leq t \leq T} I(t)$. The correctness of the ordering is evaluated by incorporating the $\tau$-Kendall measure [8]. More details on the evaluation metrics can be found in Appendix B.

The methods were applied to the following publicly available real world datasets: CollegeMsg, Facebook and Data Mining (DM) temporal citation network (refer to Appendix C for details). Each of them consists of $(u,v,t)$ tuples, where $u$ is the sender of a message or the author being cited and $v$ the receiver of the message or the author citing an article at time $t$. In order to apply $k$-shell and CI methods, we aggregate the interactions in a static unweighted graph.

4 Results

In this section, we present the results obtained for each of the datasets of Sec. 3.

4.1 CollegeMsg

We have aggregated the timestamped dataset using a one day interval. We have also applied the DR variants using time windows $dt$ of 1, 2 or 3 days.

In Fig. 2, we present $I_{max}$ for the $m = 50$ most important (top) nodes, with $\beta = 0.02$ (Fig. 2(a)) and $\beta = 0.05$ (Fig. 2(b)), keeping $\gamma = 0.01$ fixed in both cases. We observe that for $\beta$ very close to the epidemic threshold (which is calculated as $\beta_c = 0.017$ for $\gamma = 0.01$ using the method described in [22]), CI performs the best, followed by kS, DR-NF ($dt=1$), DR-NF ($dt=3$) and DR($dt=1$). DR-F behaves overall poorly. However, the case is different for $\beta = 0.05$ (Fig. 2(b)). Now, both DR-NF and DR-F are the best methods, with the latter outperforming the former. These results suggest a positive correlation between $\beta$ and the spreading efficiency of the seed sets uncovered with our method.

To investigate the preservation of the ordering for the different methods, we have used the $\tau$-Kendall measure, as described in Sec. 3. In Fig. 3, we plot the results for one parameter value per method, to make the comparisons more clear. For a complete list of the methods refer to Fig. 9 in the Appendix D. In Fig.3(a), we observe that, even though CI(l=2) has been identified as the best method for $\beta = 0.02$ and $\gamma = 0.01$, it exhibits poorer ordering preservation compared to DR-F($dt=2$). The latter is very robust within the range $10 \leq m \leq 50$. For larger values of $\beta$ (Fig. 3(b)), DR-NF($dt=2$) and DR-F($dt=2$) rapidly increase their ability to preserve the ordering as the number of the most important (top) nodes taken into account increases, closely following each other for $m \geq 30$.

In conclusion, DR-F and DR-NF variants increase their efficiency both in terms of uncovering important nodes as well as preserving the ranking ordering
Fig. 2. Plot of the $I_{max}$ for the CollegeMsg dataset, for $\gamma = 0.01$ and (a) $\beta = 0.02$ and (b) $\beta = 0.05$. The results are obtained for seed sets comprised of the $m = 50$ top nodes and are averages of 10,000 runs. The methods are indicated in the horizontal axis. In the parentheses, we include the parameters used per method.

as $\beta$ increases. Also, for the specific case of $m = 50$ most important nodes, DR-F outperforms DR-NF with respect to both $I_{max}$ and $\tau$-Kendall values.

Fig. 3. Plot of the $\tau$-Kendall as a function of the number of top nodes for the CollegeMsg dataset, for $\gamma = 0.01$ and (a) $\beta = 0.02$ and (b) $\beta = 0.05$. The methods are shown in the legends. In the parentheses, we include the parameters used per method. For a complete comparison of the methods refer to Appendix D.

4.2 Facebook

In the case of Facebook dataset, we have chosen to aggregate the timestamps in an hour interval. We have also selected the values of $dt$ to be 1, 2 and 3 hours for the DR variants.

In Fig. 4, we present $I_{max}$, for the $m = 50$ most important nodes, for fixed $\beta = 0.15$ and two values of $\gamma$ which differ by an order of magnitude. In the
case of $\gamma = 0.01$ (Fig. 4(a)), DR-NF behaves slightly better than kS, with DR-F following closely the first two methods. DR-NF continues to be the best method for the case of $\gamma = 0.001$ (Fig. 4(b)). Now, DR-F climbs in the second place and kS becomes the worst method.

![Fig. 4. Plot of the $I_{\text{max}}$ for Facebook dataset, for $\beta = 0.15$ and (a) $\gamma = 0.01$ and (b) $\gamma = 0.001$. The results are obtained for seed sets comprised of the $m = 50$ top nodes and are averages of 10,000 runs. The methods are indicated in the horizontal axis. In the parentheses, we include the parameters used per method.](image)

In Fig. 5, we plot the evolution of $\tau$-Kendall as a function of the number $m$ of top ranked nodes. For the case of $\gamma = 0.01$ (Fig. 5(a)), we observe that DR-NF(dt=2), DR-F(dt=2) and kS preserve the ranking ordering. The behavior continues to be the same for DR-F(dt=2) and kS for $\gamma = 0.001$ (Fig. 5(b)).

In conclusion, DR variants display the best performance in terms of spreading efficiency, which increases as $\gamma$ decreases. In terms of ranking preservation, only DR-F exhibits an overall stable behavior among the variants.

### 4.3 DM temporal citation network

Finally, we investigate the DM temporal citation network (see Sec. 3). We maintain the resolution of the dataset to one year and use $dt$ values of 1, 2 and 3 years for the DR variants.

In Fig. 6, we present $I_{\text{max}}$ for the $m = 50$ most important nodes. We observe that DR variants closely follow kS as the best performing method for $\beta = 0.004$ and $\gamma = 0.05$. However, all DR variants are top performing for $\beta = 0.04$ and $\gamma = 0.05$. Interestingly, DR performs better than DR-NF and DR-F for the latter set of parameters.

In Fig. 7, we monitor the behavior of $\tau$-Kendall for the different methods as a function of the number of top ranked nodes. In Fig. 7(a), we observe that kS is the best method with respect to the rank ordering preservation. DR variants are stable and have a constant $\tau$ value for $m \geq 20$, but behave worse than the kS. In Fig. 7(b), kS and DR variants are generally indistinguishable and display a relative high value of $\tau$. 
Fig. 5. Plot of the $\tau$-Kendall as a function of the number of top nodes for Facebook dataset, for $\beta = 0.15$ and (a) $\gamma = 0.01$ and (b) $\gamma = 0.001$. The methods are shown in the legends. In the parentheses, we include the parameters used per method. For a complete comparison of the methods refer to Appendix D.

Fig. 6. Plot of the $I_{max}$ for DM temporal citation network, for $\gamma = 0.05$ and (a) $\beta = 0.004$ and (b) $\beta = 0.04$. The results are obtained for seed sets comprised of the $m = 50$ top nodes and are averages of 10,000 runs. The methods are indicated in the horizontal axis. In the parentheses, we include the parameters used per method.

In terms of spreading efficiency and ranking ordering preservation, the DR variants behave better than the rest of the methods and improve their performance as $\beta$ increases, which has also been observed for the CollegeMsg dataset (Sec. 4.1).

For the DM dataset, we are also able to test our rankings against ground truth data. This is a list of the 10 most important authors on DM topic up to 2012. As we are chronologically very close to that year, it may serve as a baseline to assess the correctness of the methods used in this paper. The results are summarized in Table 2 in Appendix E. It is interesting that DR variants discover 5/10 most important authors in reality in their first $m = 10$ top ranked nodes (50% accuracy) while reaching 9/10 for $m = 50$. Thus, our method has

---

1 [http://www.kdnuggets.com/2012/03/top-10-in-data-mining.html](http://www.kdnuggets.com/2012/03/top-10-in-data-mining.html)
the ability to identify real important nodes, in conjunction with its spreading efficiency and ranking ordering preservation features highlighted earlier in this subsection.

5 Conclusions

The research on influence maximization in the field of graph theory has long focused on static representations. However, nodes and links appear and disappear through time and thus, the consideration of the temporal dimension is crucial.

In this paper, we have developed a family of methods called DepthRank (DR), able to discover important nodes in evolving networks. The results indicate that DR variants are efficient in terms of spreading and ranking ordering preservation. Using the DM temporal citation network, we were able to demonstrate that DepthRank is capable of identifying nodes that are “influential” in reality.

As a rule of thumb, we may provide the following guidelines for the application of the method: (a) $dt$ should not exceed the dataset’s resolution more than a factor of 3 and (b) the utilization of the absolute forgetting mechanism (DR-F) is the most robust.

In the future, we plan to investigate the effects of the incorporation of other types of information (text similarity, sentiments etc), as it is expected to enhance the identification of important network nodes and provide results that consider the semantic gap, as our current method takes into account only the topological interactions between users and not the interplay between topology and content.

Acknowledgement. This research was performed under the EU’s project “Trusted, Citizen - LEA collaboration over sOcial Networks(TRILLION)” (grant agreement No 653256).
Appendix A  Description of method

In time-varying networks, nodes (i.e. users in a chat forum) interact with each other at given timestamps. These interactions are not homogeneously distributed; they are denser or sparser in certain time periods, depending on various factors (i.e. type of media, occurrence of an important event etc). In the following, we proceed by aggregating these interactions using appropriate time windows with the exact values per dataset presented in Sec. 4. We denote by \( t \) the time step in the aggregated data.

After the aggregation, data are transformed into a set of directed unweighted graphs \( G_i, i = 1 \ldots M \), which are called layers. A part of them is shown in Fig. 8 (left). In this setting, there are two types of links: intra-links that connect two nodes in the same layer and inter-links that connect identical nodes in consecutive layers.

Next, we present the method for ranking important network nodes by evaluating their “influence score” \( S \). This is a two level procedure, incorporating (a) the calculation of the “influence score” update for each node in consecutive time steps and (b) the application of a monotonically decreasing function to account for the influence decay as time passes.

We denote with \( \Delta H(s,G_i) \) the “influence score” update which refers to a node \( s \) residing in layer \( G_i \). It is calculated as follows: starting from \( G_i \), we specify the number of subsequent layers \( d \) within the time range \([t_i, t_i + dt]\), with \( dt \) the (fixed) length of the time window and \( t_i \) the timestep corresponding to layer \( G_i \). Then, we construct every possible path from node \( s \) in layer \( G_i \) to any node \( w \) in layer \( G_{i+d} \), which passes through nodes in the intermediate layers (Fig. 8(right)). Note that we take into account both the inter and intra-links during path construction.

Each node \( w \) in layer \( G_{i+d} \) has \( r_{(w,G_{i+d})}^{\text{out}} \) outgoing links and \( r_{(w,G_{i+d-1})}^{\text{in}} \) incoming links in layer \( G_{i+d-1} \). We introduce the following relation:

\[
H(w,G_{i+d}|s,G_i) = \begin{cases} 
  r_{(w,G_{i+d-1})}^{\text{in}} \cdot r_{(w,G_{i+d})}^{\text{out}}, & \text{if } \text{dist}((w,G_{i+d}),(s,G_i)) < 2 \cdot d + 1 \\
  0, & \text{otherwise}
\end{cases}
\]

which couples path diversity \( (r_{(w,G_{i+d-1})}^{\text{in}}) \) and transmission efficiency \( (r_{(w,G_{i+d})}^{\text{out}}) \), to denote the portion of the update \( \Delta H \) attributed to node \( w \). Thus, \( \Delta H(s,G_i) \) is given by:

\[
\Delta H(s,G_i) = \sum_{w \in G_{i+d}} H(w,G_{i+d}|s,G_i)
\]

Referring to Fig. 8 (right), we have at layer \( i+3 \) (i is the top layer): \( H(2,G_{i+3}|1,G_i) = 2 \cdot 1 = 2 \), \( H(1,G_{i+3}|1,G_i) = 2 \cdot 2 = 4 \), \( H(3,G_{i+3}|1,G_i) = 2 \cdot 1 = 2 \), \( H(4,G_{i+3}|1,G_i) = 2 \cdot 1 = 2 \), \( H(5,G_{i+3}|1,G_i) = 1 \cdot 1 = 1 \), \( H(6,G_{i+3}|1,G_i) = 2 \cdot 1 = 2 \) and \( H(6,G_{i+3}|1,G_i) = 1 \cdot 2 = 2 \). Thus, for node \( s = 1 \) in layer \( G_i \), the update is \( \Delta H(1,G_i) = 13 \).

\( ^2 \) The number of layers \( d \) within time window \( dt \) is not fixed because the layers may not be equally spaced.
Table 1. List of notations used in the text

| Symbol | Notation |
|--------|----------|
| t      | Time step of the aggregated dataset |
| T      | Maximum time in the aggregated dataset |
| dt     | Time window for the calculation of the “influence score” update |
| Gi     | Directed unweighted graph (layer) |
| ti     | Time step corresponding to layer Gi |
| d      | Number of layers in the time range [ti, ti + dt] |
| M      | Total number of layers |
| s      | Node in layer Gi, for which “influence score” update is calculated |
| w      | Node in layer Gi + d |
| Gi+j   | The most recent layer in which node s has updated its “influence score” |
| ti+j   | Time step corresponding to layer Gi in which the “influence score” of node s has been recently updated |
| rin(w,Gi+d-1) | Number of incoming edges to node w in layer Gi+d-1 (including self-links) |
| rout(w,Gi+d) | Number of outgoing edges from node w in layer Gi+d (including self-links) |
| H(w,Gi+d|s,Gi) | Product between rin(w,Gi+d-1) and rout(w,Gi+d) for node w in layer Gi+d, which is connected through a path to node s in layer Gi |
| ΔH(s,Gi) | Update of the “influence score” of node s residing in layer Gi |
| g(ti−tis) | Forgetting function |
| S(s,Gi) | “Influence score” of node s up to layer Gi |
| S(s) | “Influence score” of node s |
| m | Number of important (top) nodes |

The “influence score” S(s, Gi) of node s up to layer Gi is the sum of the update $ΔH(s, Gi)$ and its previous value $S(s, Gi)$ in layer Gi - which corresponds to the time step $t^*_j$ of its most recent update - weighted using a forgetting function $g(t_i - t^*_j)$ to account for aging effects:

$$S(s, Gi) = ΔH(s, Gi) + S(s, Gi_j) \cdot g(t_i - t^*_j)$$

In the following, we consider three different variants: no forgetting mechanism ($g(t_i - t^*_j) = 1$), forgetting mechanism ($g(t_i - t^*_j) = e^{-(t_i - t^*_j)}$) and forgetting mechanism with normalization ($g(t_i - t^*_j) = e^{-(t_i - t^*_j)/T}$), with T the maximum time step of the aggregated dataset. Thus, the score $S(s, Gi)$, can be formulated as follows:

- No forgetting mechanism

$$S(s, Gi) = ΔH(s, Gi) + S(s, Gi_j)$$

(4)

- Forgetting mechanism:

$$S(s, Gi) = ΔH(s, Gi) + S(s, Gi_j) \cdot e^{-(t_i - t^*_j)}$$

(5)

- Forgetting mechanism with normalization:

$$S(s, Gi) = ΔH(s, Gi) + S(s, Gi_j) \cdot e^{-(t_i - t^*_j)/T}$$

(6)

The exponential decay function is commonly used in many applications, however we can replace it with any other monotonically decreasing function.
The procedure ends when we reach the last layer for which $t \leq T - dt$. The overall “influence score” $S$ for node $s$ in the underlying dataset is given by (remember that $G_j$ denotes the layer of the most recent update):

- No forgetting mechanism:
  \[ S(s) = S(s, G_j) \]  \( (7) \)

- Forgetting mechanism:
  \[ S(s) = S(s, G_j) \cdot e^{-(T - t_s)} \]  \( (8) \)

- Forgetting mechanism with normalization:
  \[ S(s) = S(s, G_j) \cdot e^{-(T - t_s)}/T \]  \( (9) \)

We finally rank the nodes in descending order according to their $S$ value. In Algorithm 1, we provide the pseudo-code for our method, including the three
variants defined previously: DepthRank (DR), where we do not impose any forgetting mechanism (eqs (4) and (7)), DepthRank with forgetting mechanism (DR-F) (eqs (5) and (8)) and DepthRank where we impose normalization (DR-NF) (eqs (6) and (9)).

Algorithm 1 Finding Important Network Nodes

\begin{algorithm}
\begin{algorithmic}
\State \textbf{Input:} timestamped dataset, \(dt\), \texttt{algType}
\State \textbf{Output:} List of \(m\) most important nodes
\begin{enumerate}
\item Import dataset
\item Aggregate dataset into layers (see main text)
\item Choose process: \(DR\) (\(\text{algType} == 0\)), \(DR-F\) (\(\text{algType} == 1\)) or \(DR-NF\) (\(\text{algType} == 2\)).
\item if \(\text{algType} == 0\) then
\item for each layer \(G_i\) do
\item for each node \(s\) in layer \(G_i\) do
\item find the paths from \(s\) in layer \(G_i\) to all nodes \(w\) in layer \(G_{i+d}\)
\item calculate eq. (4)
\item end for
\item end for
\item else if \(\text{algType} == 1\) then
\item for each layer \(G_i\) do
\item for each node \(s\) in layer \(G_i\) do
\item find the paths from \(s\) in layer \(G_i\) to all nodes \(w\) in layer \(G_{i+d}\)
\item calculate eq. (5)
\item end for
\item end for
\item else if \(\text{algType} == 2\) then
\item for each layer \(G_i\) do
\item for each node \(s\) in layer \(G_i\) do
\item find the paths from \(s\) in layer \(G_i\) to all nodes \(w\) in layer \(G_{i+d}\)
\item calculate eq. (6)
\item end for
\item end for
\item else if \(\text{algType} == 3\) then
\item for each layer \(G_i\) do
\item for each node \(s\) in layer \(G_i\) do
\item find the paths from \(s\) in layer \(G_i\) to all nodes \(w\) in layer \(G_{i+d}\)
\item calculate eq. (7)
\item end for
\item end if
\item Rank nodes according to their ‘‘influence score’’ \(S\) in descending order and output the first \(m\) nodes.
\end{enumerate}
\end{algorithmic}
\end{algorithm}
Appendix B  Baseline methods and evaluation metrics

In this section, we present the baseline methods along with the evaluation metrics. For the former, we have chosen $k$-shell ($kS$) and Collective Influence ($CI$) methods because they are widely employed in the context of graph-based techniques.

$k$-shell is broadly used for ranking purposes. Given a graph $G$, it proceeds as follows: every isolated node is considered as being in the 0-shell. For the rest of the nodes with connectivity $k \geq 1$, one first removes the links from every node with $k = 1$. The residual graph may consist of nodes with connectivity $k = 1$; thus, the same procedure is repeated until no nodes with $k \leq 1$ remain. We say that these nodes constitute the 1-shell. The rest of the shells are uncovered in the same way. The nodes belonging to the highest shells are the most important \[9\].

Collective Influence is based on optimal percolation. According to \[14\], it takes the form:

$$CI(i) = (r_i - 1) \sum_{j \in \partial Ball(i, l)} (r_j - 1)$$

where, $r_i$ is the degree of the node $i$, $\partial Ball(i, l)$ the front of a sphere centered at the $i$-th node with radius $l$ (in terms of shortest path distance). We follow the suggestions of the authors in \[14\] and set $l = 2$ and $3$ in the calculations.

To assess the spreading ability of the important network nodes uncovered by each method, we apply a temporal implementation of the SIR model, called Temporal Gillespie Algorithm \[23\] (we choose the Poisson homogeneous version), to account for the time-varying interactions in real-world networks. We have slightly modified the process in order to start from a set of initially infected nodes rather than a single one. These sets coincide with the ordered sets identified by the methods used. The efficiency of spreading is evaluated by using the maximum number of infected nodes within the whole dataset, $I_{\text{max}} = \max_{1 \leq t \leq T} I(t)$. $\beta$ and $\gamma$ are the infection and recovery rates, respectively. Except for the case of the small CollegeMsg dataset, where we have calculated the epidemic threshold for a fixed $\gamma$ value as in \[22\], for the rest of the datasets this was done heuristically, by searching for those values of $\beta$ and $\gamma$ for which considerable amount of spreading is observed. The exact values are displayed in the figure captions in Sec. 4.

We are also interested in the correct ordering of the ranking methods. A suitable measure is the $\tau$-Kendall \[8\], which evaluates the concordance of two ordered lists. We exploit it in the following way:

- We collect all the nodes tagged as important from every method $o$. We calculate $I_{\text{max}}$ for each one and rank them according to this value. This will be the overall “ground-truth” ranking list, $L$.
- We select the $m$ first nodes according to each method ranking, $R_o$, and find their position in $L$. This will be the method specific “ground-truth” ranking list, $L_o$.
- We calculate the $\tau$-Kendall between $L_o$ and $R_o$. 
This evaluation is applied for various lengths $m$ of the most important nodes lists (see respective figures in Section 4).

Appendix C  Datasets

We have performed the experiments using the following publicly available real world datasets:

- **CollegeMsg**\(^4\): It is a small temporal network corresponding to a private messaging facility at the University of California, Irvine. It consists of 1899 users and 59835 temporal edges. The timestamps are in UNIX time (seconds).

- **Facebook**\(^5\): It consists of 855542 facebook wall posts between 45813 unique users. The timestamps are in UNIX time (seconds).

- **Data Mining (DM) temporal citation network**: We have used the DBLP dataset of [21] which consists of all the papers in computer science up to 2010. The timestamps are in years.

The preparation of the DM dataset was performed as follows: we have selected the papers that contain one or more authors from a list of DM experts\(^6\). Each article record contains a set of indexes to other articles which cite the current one. We use this information to create the connections between the authors citing a paper and those of the cited paper per year of citation. For example, assume that we pick a record of the form: (title1, author1, author2, author3, 2009) which cites among other papers the following: (title2, cauthor1, cauthor2, 1993). We formulate the authors temporal interaction patterns as: (cauthor1, author1, 2009), (cauthor1, author2, 2009), (cauthor1, author3, 2009), (cauthor2, author1, 2009), (cauthor2, author2, 2009), (cauthor2, author3, 2009). Each tuple means that the first author influenced the second author in year 2009. In this way, we construct the temporal citation network for the years 1993-2010.

Each of the previous datasets comes into the form $(u, v, t)$, where $u$ is the sender of a message or the author being cited and $v$ the receiver of the message or the author citing an article at time $t$. In order to apply $k$-shell and CI methods, we aggregate the interactions in a static unweighted graph.

Appendix D  Detailed results for $\tau$-Kendall

In this section, we provide more detailed plots for the $\tau$-Kendall, by incorporating all the methods and parameters used.

\(^4\) [https://snap.stanford.edu/data/CollegeMsg.html](https://snap.stanford.edu/data/CollegeMsg.html)
\(^5\) [http://konect.uni-koblenz.de/networks/facebook-wosn-wall](http://konect.uni-koblenz.de/networks/facebook-wosn-wall)
\(^6\) [https://static.aminer.org/lab-datasets/expertfinding/datasets/Data-Mining.txt](https://static.aminer.org/lab-datasets/expertfinding/datasets/Data-Mining.txt)
Fig. 9. Plot of the $\tau$-Kendall as a function of the number $m$ of the most important (top) nodes for the CollegeMsg dataset, for $\gamma = 0.01$ and (a) $\beta = 0.02$ and (b) $\beta = 0.05$. The methods are shown in the legends.

Fig. 10. Plot of the $\tau$-Kendall as a function of the number $m$ of the most important (top) nodes for Facebook dataset, for $\beta = 0.15$ and (a) $\gamma = 0.01$ and (b) $\gamma = 0.001$. The methods are shown in the legends.
Fig. 11. Plot of the \( \tau \)-Kendall as a function of the number \( m \) of the most important (top) nodes for DM temporal citation network, for \( \gamma = 0.05 \) and (a) \( \beta = 0.004 \) and (b) \( \beta = 0.04 \). The methods are shown in the legends.

Appendix E  Comparison with ground-truth in DM temporal citation network

In Table 2, we present the number of important network nodes identified by each method that are common with those in the ground-truth list for the case of DM temporal citation network (see Section 4), as \( m \) increases.

Table 2. Number of important authors identified by the methods used compared to the top-10 DM authors list (see main text) for the DM temporal citation network.

| method                | \# of top authors |
|-----------------------|-------------------|
|                       | 10 20 30 40 50    |
| DR (dt=1)             | 5 6 7 8 8        |
| DR (dt=2)             | 5 6 7 8 9        |
| DR (dt=3)             | 5 6 8 8 9        |
| DR-F (dt=1)           | 5 6 7 8 8        |
| DR-F (dt=2)           | 5 6 7 8 9        |
| DR-F (dt=3)           | 5 5 8 8 9        |
| DR-NF (dt=1)          | 5 7 7 7 8        |
| DR-NF (dt=2)          | 5 6 7 8 9        |
| DR-NF (dt=3)          | 5 5 7 8 8        |
| CI (l=1)              | 1 1 2 2 3        |
| CI (l=2)              | 2 2 2 2 3        |
| CI (l=3)              | 1 1 2 2 3        |
| kS                    | 5 5 7 7 9        |

References

1. Aggarwal, C.C., Lin, S., Yu, P.S.: On Influential Node Discovery in Dynamic Social Networks, pp. 636–647 (2012)
2. Cai, Q., Sun, L., Niu, J., Liu, Y., Zhang, J.: Disseminating real-time messages in opportunistic mobile social networks: A ranking perspective. In: 2015 IEEE International Conference on Communications (ICC). pp. 3228–3233 (2015)
3. van Eck, P.S., Jager, W., Leeflang, P.S.H.: Opinion leaders’ role in innovation diffusion: A simulation study. Journal of Product Innovation Management 28(2), 187–203 (2011)
4. Estrada, E.: The Structure of Complex Networks: Theory and Applications. Oxford: Oxford University Press (2011)
5. Gómez-Gardeñes, J., Echenique, P., Moreno, Y.: Immunization of real complex communication networks. The European Physical Journal B - Condensed Matter and Complex Systems 49(2), 259–264 (2006)
6. Jansen, B.J., Zhang, M., Sobel, K., Chowdury, A.: Twitter power: Tweets as electronic word of mouth. J. Am. Soc. Inf. Sci. Technol. 60(11), 2169–2188 (2009)
7. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. KDD ’03, New York, NY, USA (2003)
8. Kendall, M.G.: A new measure of rank correlation. Biometrika 30(1-2), 81 (1938)
9. Kitsak, M., Gallos, L., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H., Makse, H.: Identification of influential spreaders in complex networks. Nature Physics 6(11), 888–893 (2010)
10. Laflin, P., Mantzaris, A.V., Ainley, F., Otley, A., Grindrod, P., Higham, D.J.: Discovering and validating influence in a dynamic online social network. Social Network Analysis and Mining 3(4), 1311–1323 (2013)
11. Lü, L., Chen, D., Ren, X.L., Zhang, Q.M., Zhang, Y.C., Zhou, T.: Vital nodes identification in complex networks. Physics Reports 650, 1 – 63 (2016)
12. Magnien, C., Tarissan, F.: Time evolution of the importance of nodes in dynamic networks. In: 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). pp. 1200–1207 (2015)
13. Michaelski, R., Kajdanowicz, T., Bródka, P., Kazienko, P.: Seed selection for spread of influence in social networks: Temporal vs. static approach. New Generation Computing 32(3), 213–235 (2014)
14. Morone, F., Makse, H.: Influence maximization in complex networks through optimal percolation. Nature 524(7563), 65–68 (2015)
15. Newman, M.: Networks: An Introduction. Oxford University Press, New York, NY, USA (2010)
16. Rocha, L., Masuda, N.: Individual-based approach to epidemic processes on arbitrary dynamic contact networks. Scientific Reports 6 (2016)
17. Rosas-Casals, M., Valverde, S., Solé, R.V.: Topological vulnerability of the european power grid under errors and attacks. International Journal of Bifurcation and Chaos 17(07), 2465–2475 (2007)
18. Saramäki, J., Moro, E.: From seconds to months: an overview of multi-scale dynamics of mobile telephone calls. The European Physical Journal B 88(6), 164 (2015)
19. Song, G., Li, Y., Chen, X., He, X., Tang, J.: Influential node tracking on dynamic social network: An interchange greedy approach. IEEE Transactions on Knowledge and Data Engineering 29(2), 359–372 (2017)
20. Stehlé, J., Voirin, N., Barrat, A., Cattuto, C., Isella, L., Pinton, J.F., Quaggiotto, M., van den Broeck, W., Régis, C., Lina, B., Vanhems, P.: High-resolution measurements of face-to-face contact patterns in a primary school. PLoS ONE 6(8) (2011)
21. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: Arnetminer: Extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 990–998. KDD ’08, New York, NY, USA (2008)
22. Valdano, E., Ferreri, L., Poletto, C., Colizza, V.: Analytical computation of the epidemic threshold on temporal networks. Phys. Rev. X 5, 021005 (2015)
23. Vestergaard, C., Génois, M.: Temporal gillespie algorithm: Fast simulation of contagion processes on time-varying networks. PLoS Computational Biology 11(10) (2015)
24. Viswanath, B., Mislove, A., Cha, M., Gummadi, K.P.: On the evolution of user interaction in facebook. In: Proceedings of the 2Nd ACM Workshop on Online Social Networks, pp. 37–42. WOSN ’09, New York, NY, USA (2009)
25. Zhuang, H., Sun, Y., Tang, J., Zhang, J., Sun, X.: Influence maximization in dynamic social networks. In: 2013 IEEE 13th International Conference on Data Mining. pp. 1313–1318 (2013)