1 Introduction

Most network visualization methodologies and tools focus on identifying network hubs. Hubs represent central nodes connecting sets of more peripheral nodes that are rather sparse and separate from each other, as discussed by [45]. Literature has focused on measuring centrality and provides a broad array of centrality metrics, each of them highlighting a different aspect of a hub’s prominent role. As discussed by [19], degree centrality measures the absolute number of connections of a node, closeness centrality measures how far a node is from all other nodes in the network along the overall shortest paths, while betweenness centrality assesses the role of a node as a hub of information by analyzing the extent to which the node connects separate subnetworks. These metrics represent the underlying concept of many network visualization tools. The assumption that most tools make to visualize large networks is that hubs represent the main driver of the structure of networks and, if they exist, they should be clearly highlighted to cope with complexity and obtain a nice and intuitive representation of the network.

The literature on social media makes a distinction between influencers and influence [11,30]. The former are social media users with a broad audience. For example, influencers can have a high number of followers on Twitter, or a multitude of friends on Facebook, or a broad array of connections on LinkedIn. The term influence is instead used to refer to the social impact of the content shared by social media users. The breadth of the audience was considered the first and foremost indicator of influence for traditional media, such as television or radio. However, traditional media are based on broadcasting rather than communication, while social media are truly interactive. It is very common that influencers say something totally uninteresting and, as a consequence, they obtain little or no attention. On the contrary, if social media users are interested in something, they typically show it by participating in the conversation with a variety of mechanisms and, most commonly, by sharing the content that they have liked. [8] has noted that a content that has had an impact on a user’s mind is shared. Influencers are prominent social media users, but we cannot expect that the content that they share is bound to have high influence, as discussed by [6].

In previous research, Bruni et al. [10] has shown how the content of messages can play a critical role and can be
discusses the implementation aspects of our work. Section 4 presents the experimental methodology, performance evaluation, and benchmark comparison. Conclusions are drawn in Section 5.

2 State of the Art

In this section, we will discuss about limitations of existing network visualization techniques and tools. We will also highlights the most common and widely accepted visualization aesthetic criteria.

2.1 Network Visualization Techniques and Tools

The first spring-embedded model for network visualization was proposed by [15], who have simplified the formulae used to compute spring forces, and made significant improvements by using a cooling schedule to limit nodes’ maximum displacement. However, the repulsive force was still computed between all node pairs, yielding an overall computational complexity of \(O(N^2)\) for a network with \(N\) nodes. Subsequent studies that took a similar approach are the Online Force Directed Animated Visualization (OFDAV) technique by [23], and the edge-edge repulsion approach by [34]. More recently,[44] has proposed the over relaxation algorithm for force directed drawing. Despite these efforts, these force-directed algorithms are still considered non-scalable and unsuitable for large networks, also noted by [21].

Several research efforts in network visualization have targeted power-law algorithms and their combination with the traditional force-directed techniques, as for example in [27,1]. Among these approaches, the most notable is the Out-Degree Layout (ODL) for the visualization of large-scale network topologies, presented by [38,12]. The core concept of the algorithm is the segmentation of the network nodes into multiple layers based on their out-degree, i.e. the number of outgoing edges of each node. The positioning of network nodes starts from those with the highest out-degree, under the assumption that nodes with a lower out-degree have a lower impact on visual effectiveness.

The most common and successful visualization tools are surveyed in [39,28,35] and [43]. Widely discussed tools include Cytoscape, OntoGraf, OntoSphere, GIny, graphViz, Hyper Graph, rdf Gravity, IsaViz, Jambalaya, Owl2Prefuse, Flow inspector, Gephi and SocNetV. There is no one-to-one mapping between techniques and tools. This section discusses usage results from the literature or from experimental evidence that we made with the tools.

Most of the tools are not highly scalable and with large-scale graphs, they are time inefficient or produce ambiguous layouts. Many visualization tools support graphs up to a few hundred nodes, such as rdfGravity
Gephi [41], GraphViz [16], and Flow inspector [9]. With large-scale graphs, they are time inefficient or produce ambiguous layouts, as observed by [21] with rdfGravity. Node clumping issues and edge overlap issues are common, as in Prefuse [22], Gephi [5], GraphViz, and OntoGraf [17]. Force-directed and spring layouts are implemented in several visualization tools, but local minima problems are common, as observed in SocNetV [25], Gephi, and in Flow inspector.

The most practical limitations that we have observed in existing force-directed based graph drawing techniques are the following:

- Scalability: To the best of our knowledge, most implementations scale up to few thousand nodes.
- Computational complexity: A major pitfall of existing force directed layout techniques is their computational complexity, which is $\Theta(N^2 + E)$. Hence, performance of existing approaches is low for the case of large scale networks.
- Aesthetics: Many tools suffer from node clumping and edge crossing problems in case of dense graphs, as well as vertex occlusion over edges, and asymmetric drawings as noted by [34].
- Local Minima: The adoption of cooling schedules and temperature mechanisms may reduce the problems related to local minima; however, they need to be fine-tuned and optimized to be effective on large graphs.
- Topology layout: If a network contains many edges and vertices, the structure of the visualization becomes complex due to the local minima problem.
- Convergence Nodes are moved back and forth without converging.

2.2 Influencers and Influence in Social Networks

Traditionally, the literature characterizes a social media user as an influencer on the basis of structural properties. Centrality metrics are the most widely considered parameters for the structural evaluation of a user’s social network. The centrality of a concept has been defined as the significance of an individual within a network [18]. Centrality has attracted a considerable attention as it clearly recalls concepts like social power, influence, and reputation. A node that is directly connected to a high number of other nodes is obviously central to the network and likely to play an important role [4]. [19] introduced the first centrality metrics, named as degree centrality, which is defined as the number of links incident upon a node. A node with many connections to other nodes, likely to play an important role [40]. A distinction is made between in-degree and out-degree centrality, measuring the number of incoming and outgoing connections respectively. This distinction has also been considered important in social networks. For example, Twitter makes a distinction between friends and followers. Normally, on Twitter, users with a high in-degree centrality (i.e. with a high number of followers) are considered influencers.

In addition to degree centrality, the literature also shows other structural metrics for the identification of influencers in social net-works. [31] presented an approach, where users were identified as influencers based on their total number of retweets. Results highlighted how the number of retweets are positively correlated with the level of users’ activity (number of tweets) and their in-degree centrality (number of followers). Besides structural metrics, the more recent literature has associated the complexity of the concept of influence with the variety of content. Several research works have addressed the need for considering content-based metrics of influence [7]. Content metrics such as the number of mentions, URLs, or hashtags have been proved to increase the probability of retweeting [3].

The more recent literature has associated the complexity of the concept of influence with the diversity of content. Several research works have addressed the need for considering content-based metrics of influence [32,36,42]. Clearly, this view involves a significant change in perspective, as assessing influence does not provide a static and general ranking of influencers as a result. However, there is a need for effective visualization technique in social networks, which enable user to visually explore large-scale complex social networks to identify influencers in social networks. The layout should be aesthetically pleasant and provide multi-layered periphery of the nodes in clustered networks to exploit spread of influence in social networks.

While the literature provides consolidated approaches supporting the identification and characterization of hub nodes i.e. influencers in a social network, research on information spread, which is multi-layered distribution of peripheral nodes, is limited. The literature mainly focuses on the concept of influencers, while there is a need for effective visualization techniques in social networks, which enable users to visually explore large-scale complex social networks to identify the users who are responsible for influence. This paper presents a power-law based modified force-directed technique, that extends a previous algorithm discussed in [24].

3 The Power-Law Algorithm

This section provides a high-level description of the graph layout algorithm used in this paper. An early version of the algorithm has been presented by [24]. This paper improves the initial algorithm by identifying multiple layers of peripheral nodes around hub nodes. The power-law layout algorithm belongs to the class of force-directed algorithms, such as the one by [12,20].

The base mechanism is that of starting from an initial placement of graph nodes, and then iteratively refining the position of the nodes according to a force model. The iteration mechanism is controlled by means of
CoolDown step. The main innovation in our approach consists in the synergy between the exploitation of the power-law distribution of the data and the adaptive temperature cooldown mechanism. The underlying idea is that of iterating on hub nodes first with small cooldown steps, and subsequently on peripheral nodes with large cooldown steps, in order to achieve faster convergence. The advantages of this approach are:

– The initial iteration on hub nodes is more efficient than iterating on the whole node set, since $|N_h| \ll |N|$. As a consequence, it is possible to perform a fine-grained positioning of hub nodes (achieved by adopting small cooldown steps), (peripheral nodes will then form clusters around hubs).
– The iteration over the set of peripheral nodes, which would be computationally expensive since $N_p \approx N$, is limited by the adoption of large cooldown steps.

Algorithm 1 provides a high-level overview of the whole algorithm by showing its main building blocks.

**Algorithm 1**: Abstract Level Power-Law Layout Algorithm.

Input:
- $N_h$ = Hub Nodes;
- $N_p$ = Peripheral Nodes;
- $E_h$ = Edges;
- $d$ = node’s Degree;
- $T$ = Energy / Temperature Variable;
- $T_h$ = Temperature threshold;

1. begin
2. 
call NodePartition()
3. call InitialLayout()
4. while Temperature > 0 do
5. 
  if Temperature > $T_h$ then
6. 
    call AttractionForce($N_h$, $N_p$)
7. 
    call RepulsionForce($N_h$, $E$)
8. 
  else
9. 
    call AttractionForce($N_p$, $N_h$)
10. 
    call RepulsionForce($N_p$, $E$)
11. 
end
12. call CoolDown($T$)
13. call resetNodesSizes($N_p$, $N_i$, $d$)
14. end
15. end

3.1 NodePartition

The NodePartition method is aimed at the exploitation of the power-law degree distribution of data. Provided that the degree-distribution of the nodes follows a power law, we partition the set of nodes $N$ into the set of hub nodes $N_h$ and the set of peripheral nodes $N_p$, such that $N = N_h \cup N_p$, with $N_h \cap N_p = \emptyset$. As a consequence, the set of edges $E$ is also partitioned in the set of edges $E_h$ for which at least one of the two nodes is a hub node, and the set $E_p$ which contains all the edges connecting only peripheral nodes, with $E = E_h \cup E_p$, and $E_h \cap E_p = \emptyset$. The distinction of a node $n$ as a hub node or as a peripheral node is based on the evaluation of its degree $\rho(n)$ against the constant $\rho_h$, which is a threshold defined as the value of degree that identifies the top $i^{th}$ percentile of nodes, sorted by decreasing value of degree. Since the power-law is supposed to hold in the degree distribution, assuming for example $i = 20$ will end up in defining $\rho_h$ as the $20^{th}$ percentile, thus considering as hub nodes the $20\%$ of the nodes with the highest values of degree - the Pareto’s 80-20 Rule, as suggested by [29].

3.2 InitialLayout

The InitialLayout() method responsible for random placement of graph nodes. However, as discussed by [13] and [27], it is known from the literature that the initial layout of graph nodes is an important factor to be considered in order to avoid the local minima problem, especially as the number of graph nodes increases, as noted by [27, 15]. As suggested by [14], a combined approach can be helpful in solving this problem. In this paper, we adopt a random initial placement of nodes; however, a combination with other algorithms such as [26] or [20] will be considered as part of our future work.

[27] The initial iteration on hub nodes is more efficient than iterating on the whole node set, since $|N_h| \ll |N|$. As a consequence, it is possible to perform a fine-grained positioning of hub nodes (achieved by adopting small cooldown steps), (peripheral nodes will then form clusters around hubs).

[28] The iteration over the set of peripheral nodes, which would be computationally expensive since $N_p \approx N$, is limited by the adoption of large cooldown steps.

3.3 Forces

In this paper, both forces formulae (Attraction and Repulsion) have been taken from the power-law based modified force-directed algorithm as presented in [24].

3.4 CoolDown

The CoolDown($T$) method is responsible of cooling down the system temperature, in order to make the algorithm converge. We introduce a customized dynamic temperature cooldown scheme, which adapts the cooldown step based on the current value of the temperature. As shown in Figure 1, the temperature is supposed to be initialized at a value $T_{start}$, and then to be
Algorithm 2: Temperature Cooldown

```
1 begin
2   if Temperature > T_h then
3     Temperature = Temperature - Δt_h;
4   else
5     Temperature = Temperature - Δt_p;
6   end
7   if Temperature ≤ T_c then
8     Temperature = 0;
9 end
```

This method is responsible for resetting the sizes of each node in the graph, based upon their degree. The higher the degree of a node, the greater the size and vice versa.

3.6 Computational complexity

We evaluate the overall computational complexity of the graph layout algorithm by starting from the assessment of the computational complexity of its components.
Two languages have been considered, English and Italian. Collected tweets have been first analysed with a proprietary semantic engine in order to tag each tweet with information about (a) the location to which it refers, (b) the location’s brand driver (or category) on which authors express an opinion, (c) the subject referred to by the author, (d) the number of retweets (if any), and (e) the identifier of the retweeting author. Our data sample is refers to the tourism domain. We have adopted a modified version of the Anholt Nation Brand index model to define a set of categories of content referring to specific brand drivers of a destination’s brand [2]. Examples of brand drivers are Art & Culture, Food & Drinks, Events & Sport, Services & Transports, etc. A tweet is considered Generic if it does not refer to any Specific brand driver, while it is considered Specific if it refers to at least one of Anholt’s brand drivers.

Tweets have been categorized by using an automatic semantic text processing engine that has been developed as part of this research. The semantic engine can analyse a tweet and assign it to one or more semantic categories. The engine has been instructed to categorize according to the brand drivers of Anholt’s model, by associating each brand driver with a specific content category described by means of a network of keywords. Each tweet can be assigned to multiple categories. We denote with \( N_C \) the number of categories each tweet \( w \) is assigned to; the specificity \( S(w) \) of a given tweet \( w \) is defined in Equation 1 as follows:

\[
S(w) = \begin{cases} 
0, & N_C = 0 \\
1, & N_C > 0 
\end{cases}
\]

Table 1 refer to the descriptive statistics of the original non-linear variables.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of tweets</td>
<td>957,632</td>
</tr>
<tr>
<td>Number of retweeted tweets</td>
<td>79,691</td>
</tr>
<tr>
<td>Number of tweeting authors</td>
<td>52,175</td>
</tr>
<tr>
<td>Number of retweets</td>
<td>235,790</td>
</tr>
</tbody>
</table>

4.2 Network models

In order to verify the effectiveness of the proposed algorithm with respect to the goal of our research, we have defined different network models based on the data set described in the previous section. Figure 2 provides an overview of the adopted network models.

- **Author → Brand \((N_1)\)** This model considers the relationship among authors and domain brands, i.e., touristic destinations in our data set. The network is modeled as an undirected affiliation two-mode network, where an author node \( n_a \) is connected to a brand node \( n_b \) whenever author \( a \) has mentioned brand \( b \) in at least one of his/her tweets. The weight of the edge connecting \( n_a \) to \( n_b \) is proportional to the number of times that author \( a \) has named brand \( b \) in his/her tweets.

- **Author → Category \((N_2)\)** This model considers the relationship among authors and domain brand drivers (categories), i.e., city brand drivers in our data set (namely, Arts & Culture, Events & Sports, Fares & Tickets, Fashion & Shopping, Food & Drink, Life & Entertainment, Night & Music, Services & Transports, and Weather & Environmental). The network is modeled as an undirected affiliation two-mode network, where an author node \( n_a \) is connected to a category node \( n_c \) whenever author \( a \) has mentioned a subject belonging to category \( c \) in at least one of his/her tweets. The weight of the edge connecting \( n_a \) to \( n_c \) is proportional to the number of times that author \( a \) has named category \( c \) in his/her tweets.

- **Author → Subject \((N_3)\)** This model considers the relationship among authors and domain subjects, i.e., relevant semantic lemmas in our data set. The network is modeled as an undirected affiliation two-mode network, where an author node \( n_a \) is connected to a subject node \( n_s \) whenever author \( a \) has mentioned subject \( s \) in at least one of his/her tweets. The weight of the edge connecting \( n_a \) to \( n_s \) is proportional to the number of times that author \( a \) has named subject \( s \) in his/her tweets.

- **Author → Author \((N_4)\)** This model considers the relationship among authors producing a tweet and corresponding retweeting authors. The network is modeled as a directed one-mode network, where an author node \( n_{a1} \) is linked to another author node \( n_{a2} \) whenever author \( a1 \) has retweeted at least one tweet of author \( a2 \). The weight of the edge connecting \( n_{a1} \) to \( n_{a2} \) is proportional to the number of times that author \( a1 \) has retweeted author \( a2 \).

4.3 Visualization Results and Discussions

In order to visually analyse the influencers (hub nodes) and influence (spread across the multi-layered peripheral nodes connected around hub nodes), we visualized aforementioned networks in Section 4.2. The color scheme for node-pair for all networks, is consistent for each graph (Yellow nodes: \( N_4 \); Blue: \( N_0 \)). Figures 4, 6, 7 and 9 present visualizations of the each network \((N_1 → N_4)\) from dataset, along with visual benchmark comparison with existing approaches. Table 2 compares the average time performance of our algorithm against that of the [20] and [33] approaches. Our approach shows a significant improvement in layout computation time.
Table 2: Summary of experimental results.

<table>
<thead>
<tr>
<th>Dataset Size</th>
<th>Computational Time and Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PL</td>
</tr>
<tr>
<td></td>
<td>(s)</td>
</tr>
<tr>
<td></td>
<td>(%)</td>
</tr>
<tr>
<td>$N_1$</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>0.012</td>
</tr>
<tr>
<td>275</td>
<td>1.223</td>
</tr>
<tr>
<td>2,627</td>
<td>4.962</td>
</tr>
<tr>
<td>12,017</td>
<td>6.472</td>
</tr>
<tr>
<td>21,000</td>
<td>9.635</td>
</tr>
<tr>
<td>30,523</td>
<td>12.256</td>
</tr>
<tr>
<td>$N_2$</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>0.0107</td>
</tr>
<tr>
<td>87</td>
<td>0.114</td>
</tr>
<tr>
<td>301</td>
<td>1.236</td>
</tr>
<tr>
<td>2,659</td>
<td>3.248</td>
</tr>
<tr>
<td>12,049</td>
<td>7.623</td>
</tr>
<tr>
<td>$N_3$</td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>0.025</td>
</tr>
<tr>
<td>583</td>
<td>1.367</td>
</tr>
<tr>
<td>3,694</td>
<td>2.923</td>
</tr>
<tr>
<td>$N_4$</td>
<td></td>
</tr>
<tr>
<td>1,305</td>
<td>0.941</td>
</tr>
<tr>
<td>2,677</td>
<td>1.769</td>
</tr>
<tr>
<td>6,268</td>
<td>2.746</td>
</tr>
<tr>
<td>11,484</td>
<td>4.627</td>
</tr>
</tbody>
</table>

Key to symbols: $N$: total number of nodes in network; $N_A$: number of author nodes; $N_B$: number of brand / subject / category / retweeting author nodes; $E$: number of edges

Key to algorithm acronyms: PL: Power-law; FR: Fruchterman-Reingold; MS: Modified Spring.

The dataset follow a power-law distribution, as discussed by [37]. Figure 3 explains that the graphs in our test set are ‘scale-free’ as they exhibit power-law degree distribution.

4.3.1 Results – $N_1$ Network (Brand Fidelity)

Networks $N_1$ is related to the relationship between authors and brands, i.e., touristic destinations which are basically Italian cities. In this case, the clustering of nodes provides a grouping of those authors who have tweeted about the same destination. The layering of nodes around brands is instead related to the intensity of tweeting about a given destination; i.e., authors closer to a brand node tweet a higher number of times about that destination with respect to farther authors. The emerging semantic of the network visualization is in this case related to the Brand Fidelity of authors. The visualized network layout supports the visual analysis of those authors who have a higher fidelity to a given brand, or those authors who never tweet about that brand. Moreover, it is possible to point out which authors are tweeting about a brand as well as a competing brand to support the definition of specific marketing campaigns. Through our visualization approach, we are able to visually identify multiple peripheral layers of nodes surrounded by influencing hub nodes, the spread of these multi-layered peripheral nodes around hub nodes express the influence. Figures 4 provides the visualization of networks $N_1$ of our dataset, together with a visual comparison with the layouts generated by two
clearly highlights that author nodes aggregate in several groups and subgroups based on their connections with category nodes, which in this case are the hub nodes. The aggregation of author nodes can be analyzed from two different perspectives:

1. Clusters. The groups of author nodes cluster together all those authors that are connected to the same hubs (i.e., categories); this provides a visual clustering for those authors who have tweeted about the same categories. For example, Figure 5 highlights clusters that group all the authors who tweeted about *Events & Sports, Fashion & Shopping, Drink*, and *Entertainment* categories, as well as the authors who tweeted about more than one category, such as *Transport and College, or Entertainment and Photo*.

2. Layers. The network layout shows that clusters are placed at a different distance from the visualization center based on the number of hubs to which they are connected. In other words, the most peripheral clusters are those in which nodes are connected to only one hub, while the central cluster is the one in which nodes are connected to the highest number of hub nodes. An example of node layering is provided in the upper left area of Figure 5: the cluster referring to those authors who have tweeted about category *Entertainment* is positioned above (i.e., on an outermost layer) and the clusters grouping the authors who have tweeted about *Entertainment* and *Photo*, or *Entertainment* and *People* are positioned below.

Authors belonging to the central cluster of nodes are in fact those who are more *generalist* in their content sharing about the analyzed tourism destinations, since they refer to many different categories. On the contrary, authors belonging to the most peripheral clusters are those who are very *specific* in sharing content related to selected categories. Figure 6 represents the benchmark comparison of our technique with existing techniques, and the results are evident that our approach produces aesthetically pleasant layouts by highlighting clusters of multiple peripheral layers surrounded by hub-nodes.
4.3.3 Results – $N_3$ Network (Subject Specificity)

Network $N_3$ is related to the relationship between authors and subjects. Figures 7 provides the visualization of networks $N_3$ of our dataset, together with a visual comparison with the layouts generated by two reference algorithms. The emerging semantic of the network visualization is similar to that of $N_2$, since the layout provides a visual representation of the level of specificity (or generality) of authors with respect to subjects instead of categories. In this network, we found many subjects, upon which multiple authors expressed their opinions, hence the center of graph, seems dense.
Our approach able to produce multiple layers of peripheral nodes surrounded by hub-nodes. In graph, we can observe multiple outlier peripheral layers, which are surrounded by distinct subjects, are drawn far from center of graph. We also observe some influencing authors’ nodes of large size, as they seemed to express their opinions many times upon multiple subjects, hence showing strong influence.

Fig. 7: Network $N_3$: Author $\rightarrow$ Subjects $G(N=3,694;E=10,489)$.

4.3.4 Results – $N_4$ Network (Retweeting phenomena)

Network $N_4$ is related to the relationship among authors retweeting other authors. Although very simple, this network model visually represents the complexity of real-world retweeting phenomena. As depicted in Figure 8, different retweeting scenarios are associated with different network topologies.

1. *Cloud Retweeting*: In case a) of Figure 8, an author is retweeted by many of his followers, is visually represented as a cloud of nodes aggregating around a single hub.

2. *Chain Retweeting*: The opposite situation, depicted in case c) of Figure 8, is that of a tweet that is retweeted by an author which is following the author who has last retweeted.

3. *Mixed Topology*: In the middle, as represented by case b) of Figure 8, a combination of the two base scenarios may happen, leading to intermediate topologies of varying complexity.

For Network $N_4$ visualizations which are provided in Figure 8, Figure 9 and 10, a specific node coloring scheme is adopted in order to distinguish among different types of authors. Yellow nodes represent those authors who only retweets other authors, and Blue nodes represent those authors who only retweeted by other authors. Similarly, Green nodes represent authors who both retweet and retweeted by other authors.

Figure 9 represents the benchmark comparison of our technique with existing techniques. By considering only hub nodes, in fact, it is clear that there is no clue to understand how content spreads across the authors network, since the majority of hubs are just the centers of isolated clouds of authors. Interesting insights can be provided to the reader only by taking into account the peripheral nodes (i.e., those nodes that are not labeled as hubs), and thus by reconstructing the phenomenon of chain retweeting. The network layout generated by the proposed power-law layout algorithm is clearly effective in helping the reader in identifying the different retweeting scenarios and interpreting how retweets spread across the network of authors.

Fig. 8: Examples of author-author retweeting scenarios: a) cloud retweeting; b) mixed topology; c) chain retweeting.

Fig. 9: Network $N_4$: Author $\rightarrow$ Author $G(N=2,677;E=2000)$

The interesting retweeting scenarios are the chain retweeting ones as shown in Figure 10. By considering only hub nodes, in fact, it is clear that there is no clue to understand how content spreads across the authors network, since the majority of hubs are just the centers of isolated clouds of authors. Interesting insights can be provided to the reader only by taking into account the peripheral nodes (i.e., those nodes that are not labeled as hubs), and thus by reconstructing the phenomenon of chain retweeting.

5 Conclusions and Future Work

This paper proposes a novel visual aspect for the analysis and exploration of social networks in order to identify and visually highlight influencers (i.e., hub nodes), and
influence (i.e., spread of multi-layer peripheral nodes), represented by the opinions expressed by social media users on a given set of topics. Results show that our approach produces aesthetically pleasant graph layouts, by highlighting multi-layered clusters of nodes surrounding hub nodes (the main topics). These multi-layered peripheral node clusters represent a visual aid to understand influence.

Our approach exploits the underlying concept of power-law degree distribution, which effectively represent multi-layered peripheral clusters around hub nodes. We analysed four different networks to exploit brand fidelity, category specificity, subject specificity and retweeting phenomenon. Our proposed approach is able to handle scalable graphs in multi-clustered, and multi-layered peripheries of network and encourages us to further explore social network’s intrinsic characteristics. Results show that our approach significantly improves scalability, time performance and visual effectiveness compared to previous approaches. Although our experiment can be repeated with data from entities different from tourism domain, additional empirical work is needed to extend testing to multiple datasets and domains.

Future work will consider influence-based exploration of social networks based on influential parameters. An empirical evaluation of generally accepted graph drawing aesthetics criteria can be considered, to compare our approach with existing network drawing techniques. In our current work, we are studying an achievable measure of influence through proposed visualization approach that
can be used to rank influential nodes in social networks. Future research may address the development of an ad-hoc tool, by using proposed technique, for influence-based exploration of social networks.

References


