UNIVERSIDAD CARLOS III DE MADRID

Departamento de Matemáticas

MASTER'S THESIS

New Methods for Ranking Influence in Social Networks

Luis Felipe Núñez Chiroque

Tutor: Ángel Sánchez Sánchez

February, 2015

Acknowledgments

I would like to thank both my supervisor Dr. Antonio Fernández Anta and my tutor Dr. Anxo Sánchez Sánchez for all their help and collaboration in the realization of this Master's Thesis, as well as I would thank Dr. Agustín Santos Méndez and Dr. Christopher Thraves Caro.

I also thank each and every one of the teachers I have had throughout my training in this Master from whom I have always learned something.

Thank you to my colleagues, with whom I have been progressing side by side, helping and supporting each other.

And last but not least, I need to thank my family, all of them, especially my parents, who always have given me all the help and support I ever needed.

Agradecimientos (*Spanish*)

Me gustaría agradecer tanto a mi supervisor Dr. Antonio Fernández Anta como a mi tutor Anxo Sánchez Sánchez por toda su ayuda y colaboración en la realización de esta Tésis de Máster, así como también agradecerles al Dr. Agustín Santos Méndez y al Dr. Christopher Thraves Caro.

También quiero agradecer a todos y cada uno de los profesores que he tenido durante mi formación en este Máster, de los cuales siempre he aprendido algo.

Agradecer a mis colegas, con quienes he ido progresando codo con codo ayudándonos y apoyándonos los unos a los otros.

Y por último y no por ello menos importante, tengo que dar las gracias a toda mi familia, especialmente a mis padres, quienes siempre me han ayudado y he contado con su apoyo cuando lo he necesitado.

Preface

This work agrees to the guidelines of the **A-Modality**, described below.

A-Modality Original research work which solves a research open problem in a knowledge area related with Master's thematic.

Social Network Analysis is a wide interdisciplinar research field which combines both Social and Natural Science. In the last years, because of the emergence of Online Social Networks (OSN's; e.g.: Facebook, Twitter, etc.), new data is generated every minute over the whole world. This large flow of continuously arriving data opens new interesting problems. Dealing with the dynamics and time effects of these huge networks is a challenge for scientists.

New Methods for Ranking Influence in Social Networks

Luis F. Chiroque

Carlos III University of Madrid
28918 Leganés
Madrid, Spain
100302819@alumnos.uc3m.es

Abstract

In this work, propagation dynamics on social networks are studied in order to identify the most influential users. For this purpose, diffusion data has been collected during 4 weeks from a microblogging OSN (online social network) called *Tumblr*[1]. Then, the propagation graph has been built and studied using the first 2 weeks data (period T_1). Subsequently, this graph has been used to predict the influencers during the last 2 weeks (period T_2). A ranking of influential nodes is obtained for T_2, set as the *ground truth*. The aim is to predict this ranking using the data from T_1. Based on the average spread of users' posts, rankings obtained with several techniques are tested and compared. These techniques include classical centrality measures used in the literature, the T_1 ranking itself, and new alternatives based on *effective degree* using local (network) information. Whilst all methods perform similarly when considering whole global ranking, differences among them appear when ranking the top influencers. For those, in general, the methods proposed here outperform the classical centrality measures.

1 Introduction

In the last years, epidemics and information spreading in complex networks have been widely studied from a social-network point of view. Studying the role of some important nodes in the spreading dynamics may be useful for understanding and controlling these processes. Whilst the propagation of epidemic diseases is studied for minimizing the impact of such diseases (e.g., vaccinating/removing central nodes) [1], rumor spreading processes aim at maximizing the propagation of such rumors (news, pieces of information, opinions) [13] or minimizing them (worms, viruses) [19].

On microblogging OSNs (online social networks) the dynamics of the information spreading work as follows. Users (the network nodes) can publish content (text, media, etc.); this action is called *posting* and the content is a *post*. Also, users can subscribe to other users' content, by *following* these users, which is an unilateral action (e.g., $user_1$ follows $user_2$ without any action or acknoledgment from $user_2$). Thus, every time a user posts something, it is broadcasted to all his/her followers. They can decide to *re-post* such content, which will be broadcasted to their followers as well. This way, cascade reactions might be generated.

Thus, classically, a directed graph generated from the users' follower list is studied, in order to detect "influencers." These are nodes that have a good connectivity and/or good position within the network to generate such large information cascades.

[1] All data has been collected using the oficial API provided by Tumblr Inc.

In this line of research, the first problem to be solved is how to define and measure the influence of a user. Some authors have suggested defining influence for a user u as the probability of finding a re-post, during a period of time, which was originally posted by u, normalized by the total amount of posts (and re-posts) during such period [7]. However, this may be criticized, as it is not using the network information. Then, more recent studies have proposed to use the average cascade size that users create over a period of time [2].

Second, users' influence has to be sorted somehow yielding a *ranking*. Multiple studies in the literature have applied network centrality or link analysis metrics [4, 9, 21] to rank the nodes by their relevance. However, none of them combines these techniques with actual diffusion data. This is important because diffusion data may unveil active areas in the network, while reducing the relevance of the node degrees (followers). This is important because the amount of followers is not a reliable influence metric [7]. Thereby, the amount of *active* followers a user has is more relevant than the total amount of them. Using number of followers only may distort the results. Moreover, when dealing with large graphs, the use of classical centrallity metrics may have a high computational cost and thus, other methods have been recently proposed. For instance, they randomly choose potencial high-degree central nodes [10], but for other purposes.

As a result of all these studies, which characterize the spreading dynamics on OSNs, new models have arised which are able to capture the network effects in short-time periods with reasonable good results [12, 18].

1.1 Motivation

Since most of recent studies are focused on Twitter as the reference OSN, it is desireable to contrast the results obtained with other OSNs in order to generalize the conclusions. If the results are not biased by Twitter data's nature, similar results should be obtained in any OSN. Thus, the microblogging OSN *Tumblr* [8] has been chosen in this work for this purpose. Tumblr has the nice property that, unlike in Twitter, propagation cascades can be exactly obtained. As a drawback, we cannot get the whole followers lists, but this should not be a problem according to the *follower fallacy*[2] [7], as long as we can get the active followers.

Relying on actual diffusion data is a key factor to verify the contribution of the network dynamics (user-to-user interactions) on the influence rankings, as well to justify their use. We aim in this paper to ranking the nodes on a network using past diffusion data, and check our ranking with future diffusion data. The proposed ranking algorithms should be easily computable, scalable, and must use both diffusion and network information, if possible. Further, we will try to draw conclusions for better understanding OSN dynamics in order to improve existent propagation models.

1.2 Contributions

In this paper we make the following contributions; (i) We have used a real diffusion dataset for our experiments. The evaluation of real data has a major relavance for supporting theoretical results which rely on synthetic graphs and/or simulations. (ii) We have built a diffusion network using actual data. In this network arc weights are computed as the re-post rate. (iii) We have proposed new techniques for ranking nodes by influence in a diffusion network. (iv) Actual diffusion data has been used for evaluating the prediction capacity of different tehcniques. This capacity has been measured by having the techniques generating ranking of nodes by influence. The techniques include:

[2]It has been believed that *the more followers you have, the more influent you are*, which is proved to be a fallacy.

- Past diffusion data. The average propagation of diffusion trees are estimated and these tree sizes define a ranking.
- Classical centrality metrics, each of which provides a ranking.
- The techniques proposed here.

(v) With one exception, the techniques proposed here for ranking nodes have outperformed the others. (vi) We have verified that some of the properties observed in Twitter happen in Tumblr as well, such as the oscillation of influencers over time or the re-blog delay distribution.

1.3 Structure

In Section 2 we describe the different influence metrics used in the literature, and discuss their use and relevance, providing alternative solutions. The Tumblr dataset description, retrieving method, and details can be found in Section 3. The results and comparison among the different influence rankings are shown in Section 4. Finally, we present our conclusions and future work in Section 5.

2 Influence Metrics

We are going to define the different rankings that will be used in this work. First of all, we are going to define any *network* as a *graph* $G(V, E)$, where V is the set of nodes in the network and $E \subseteq V \times V$ the arc set. We are not considering *loops* in this graph definition (i.e.: $\nexists (u, v) \in E$ such that $u = v$). We assume *weighted graphs* where each arc has associated a *weight* defined by a function $weight : E \longrightarrow [0, 1]$. Without loss of generality, any *unweighted* directed graph can be seen as a weighted graph where all its arc weight equally. Moreover, in some cases, uniform-weighted graph (a graph with its arcs having the same weight) might be equivalent to an unweighted graph, using the former definition.

We have to do an important observation with the graph arc definition meaning. The arc direction for any arc $(v, w) \in E$ means that the arc goes from v to w because the information propagation goes from v to w. This is different from previous works where the arc would go from w to v because w is following v, even when the information goes in the opposite direction. Thus, we define the following subsets for describing the vertex *neighbourhood*.

$$N_{in}(v) = \{u \in V : \exists (u, v) \in E\}, \text{ and } N_{out}(v) = \{u \in V : \exists (v, u) \in E\}. \quad (1)$$

Recalling that $(u, v) \neq (v, u)$, and $u \neq v$ (no *loops*), we say that $N_{in}(v)$ is the set of nodes that v is following (v-*followees*), and $N_{out}(v)$ is the set of nodes that are following v (v-*followers*).

A *ranking* will be a vertex permutation supported by a rank function f defined as any function $f : V \times \mathcal{G}(V) \longrightarrow [0, \mathcal{O}(|V|)]$, where V is the set of nodes in the network, $\mathcal{G}(V)$ the set of graphs with vertex set V, and $\mathcal{O}(|V|)$ means a linear value with the number of vertices. Note that applying this function over the vertex set V gives a partially ordered set which allows to sort the nodes by its ranking value (numeric). Although the metrics we will present here are not explicitly using the graph as a funtion argument they will use it somehow through the adjacency matrix, the short-paths, local connectivity, etc.

2.1 Centrality Metrics

Next, we are going to describe each centrality metric used in this work, classified by its type as follows. Unless we say otherwise, this metrics has a weighted version which will be used in our tests.

Degree The most basic centrality metric for any vertex v is the degree $\kappa(v)$ defined as the number of arcs leaving v, i.e. $|N_{out}(v)|$. Since we are not able to obtain the actual followers lists in *Tumblr*, our degree will be computed considering those followers that have showed activity (see Section 3.2). This metric has no weighted version.

Betweenness This metric quantifies how many times a node v appears in short-paths between pairs of nodes $(u, w) \subseteq V \times V$. It is defined in [16] as follows. For any vertex v, it is defined as

$$B_v = \sum_{u,w \in V, u \neq w} \frac{\sigma(u, v, w)}{\sigma(u, w)}, \tag{2}$$

where $\sigma(u, v, w)$ gives the number of shortest-paths between u and w that go through v, and $\sigma(u, w)$ gives the total amount of shortest-paths between u and w. The *weighted* version is obtained by using Dijkstra's distance algorithm when computing the shortest paths [17]. The distance between two connected vertices u, v is given by the weight inverse $\frac{1}{weight((u,v))}$.

Closeness This metric [5] is defined as follows.

$$C_v = \frac{|V|}{\sum_{w \in V, w \neq v} d_{vw}}. \tag{3}$$

It considers the shortest distance a vertex has to each other vertex in the network. The longer the distance, the higher the denominator. The *weighted* version is obtained by using Dijkstra's distance algorithm when computing the shortest paths [17], as well as in *betweenness*. Thus, the higher the weight, the shortest the distance.

μ-PCI This metric is proposed in [4] for detecting influential spreaders in complex networks. It is a metric defined for unweighted graphs, which has been proposed for outperforming existent metrics such degree and *k-shell* decomposition [14] and it is defined as follows. A vertex v is assigned a number k such that no more than μk nodes in the μ-hop neighbourhood of v has degree greather or equal to k, where the μ-hop neighbourhood is the subgraph centered in the vertex v that contains all the vertices (and their arcs) that are reachable using at most μ arcs from v. As in [4], we will use $\mu = 1$. This metric is expected to outperform the ranking generated by the degree as claimed, and will be a reference for other metrics that rely on unweighted graphs.

2.2 Link Analysis Metrics

PageRank Google PageRank is a metric used for ranking documents in hyperlink networks [6]. It measures the probability of reaching a node in the network, if we start from any other node following a random travel through the network. It is computed as the vector $\pi = \pi G$, where G is the *Google* matrix defined as

$$G = \alpha(P + \frac{ae^T}{N}) + \frac{(1-\alpha)}{N} ee^T, \tag{4}$$

where α is the probability of randomly jumping to any other node in the network at any moment (usually set to 0.85), $a \in [0, 1]^N$ is called dangling value vector, $e \in [1]^N$, and $P(i, j) = \frac{A_{ij}}{\sum_k A_{ik}}$, where A_{ij} are the elements of the adjacency matrix[3]. The *weighted* version is computed using the adjacency matrix of the weighted graph [20].

[3]Since our network arcs has the flow information direction, for computing PageRank, we have used the transposed adjacency matrix A^T for changing the arcs direction in this particular case.

HITS The HITS algorithm gives two values for each network vertex (*authority* and *hub* scores) [15]. It is defined as two vectors, x and y, such that the *authority* scores x can be determined from the hub scores $x = A^T y$ and the *hub* scores y from the authority scores $y = Ax$. Summing up, they are the singular vectors

$$x = A^T Ax \qquad \text{and} \qquad y = AA^T y. \tag{5}$$

Unlike pageRank, it is not necessary transpose the adjacency matrix A in this case (see the *footnote*), but then authority and hub scores will be exchanged in comparison with the follower direction arc networks. For this reason, we expect hub scores outperform the authorithy scores when detecting influent nodes. The *weighted* version is computed using the adjacency matrix of the weighted graph [20], as well as in *PageRank*.

2.3 New Approaches

In view of previous work results, we realize that the vertex degree might not capture the actual network dynamics (activity). Thus, it would be desirable replacing the actual vertex degree (out-degree, i.e.: the number of followers) by a more meaningful metric. Moreover, in this particular case we are not sure the actual degree values, since *Tumblr* does not allow to obtain the followers lists. For this purpose, we are going to define the *effective degree* similarly as vertex *strength* is defined [3].

Definition 1 *The* effective degree *of a vertex v is defined as* $\hat{\kappa}_v = \sum_{w \in N_{out}(v)} weight((v,w))$,

where $N_{out}(v) \subseteq E$ is the set of arcs that leave v. Note that $\hat{\kappa} \in [0, \kappa]$. In Section 3.2 it is explained how weights are computed from difussion data and it how they measure the activity among pair of users. Recall that for unweighted graphs, *weight* function returns 1 and the definition is equivalent to degree. The effective degree might be a metric by itself, and actually, we will test it in our performance evaluation. However, we are looking for a metric which takes the network into account. Then, we define the following metric.

Definition 2 *The* 1-hop average effective degree *is defined as*

$$\tilde{f}(v) = \hat{\kappa}_v \frac{\sum_{w \in N_{out}(v)} \hat{\kappa}_w}{\kappa(v)} \tag{6}$$

The second term is the average of the out-neighbours effective degree. Intuitively, we are averaging the effective degree for the 1-hop neighbourhood and it might be generalized for a μ-hop neihbourhood. Nevertheless, the degree term still appears.

Ego Additive Effective Degree This metric is not degree dependent and just use its effective degree and his followees' (ego network). It is defined for each vertex as follows.

$$f(v) = \tilde{f}(v)\kappa(v) = \hat{\kappa}_v \sum_{w \in N_{out}(v)} \hat{\kappa}_w \tag{7}$$

So, multiplying the *1-hop average effective degree* of a vertex by its actual degree yields this measure.

3 Data Set

The data set we have used for our experiments is described in this section. Our first step is to select the portion of the social network we are going to work with. The next step is to collect all the difussion activity our network generated during the period of time we are going to study (4 weeks).

3.1 Selecting our sub-Network

As suggested in [7] it is interesting, but not necessary, to study the influence of users that share content related with the same topic. In this line, we have tried to get users which are related through common interests. Besides, as the same study claims, some users try to get popularity during some key events, something we will take into account for the results.

In order to capture all these features, we have selected a famous sporting event, the *2014 UEFA Champions League Final*. We have called the time of the event t_{ev}, and we define two periods. The period $T_1 = [t_{ev} - 2\ weeks, t_{ev})$ as the *past*, and the period $T_2 = [t_{ev}, t_{ev} + 2\ weeks)$ as the *future*. Additionally, we will define $T = T_1 \cup T_2$ as the whole observed period. Then, using the Tumblr API[4] feature of looking for tagged content, we retrieve all posts tagged with "champions-league" during T. That gives us a total of 872 different users who generated content (posts) related to this topic. These are the users we are going to rank. Next, using the Tumblr API as well, we obtained all the propagation cascades (reblog/re-post cascades) for each single post, and we aggregated all the cascades, yielding the network to study. Note that in this process, we are discovering new nodes at each cascade flow, aside from the initial 872. In this network each user on the cascades is a node (set V), and the information cascade flow among this users (the re-posts) determine the sense and direction of the arcs (set E). Our network is a graph of 17,756 vertices and 205,011 relationships (14% of reciprocity, i.e., nodes linked bi-directionally), which determines a out-degree's power-law exponent of 2. This network comprises a large connected component (98.1% of the vertices), singletons (1.2%) and the remaining vertices are spreaded in smaller components. Further information about general *Tumblr* stats can be found in [8], where it is compared with general stats from other OSNs as well.

3.2 Obtaining a Weighted Network from Diffusion Data

The last step is obtaining all the local activity each node had during T_1. The local activity of a node is defined for each out-arc. For any arc $(v, w) \in E$, its $weight((v, w))$ is computed as

$$weight((v,w)) = \frac{\#\text{re-posts}_{vw}^{T_1,\tau}}{\#\text{posts}_v^{T_1'} \bigcup \sum_{u \in N_{in}(v)} \#\text{re-posts}_{uv}^{T_1'}}, \tag{8}$$

where $\#\text{re-posts}_{vw}^{T_1,\tau}$ is the number of re-posts the vertex w does of content published by v during T_1 with a delay of at most τ (analogous for $\#\text{re-posts}_{uv}^{T_1'}$, but without any τ constraint), and $\#\text{posts}_v^{T_1'}$ is the number of posts that v published by itself. Simplifying $T_1 = [t_i, t_f)$, we have defined $T_1' = [t_i, t_f - \tau)$ used in the denominator. Thus, all posts have the same timeout τ to consider its re-posts, which is fair along all nodes. Note that, since $T_1' \subseteq T_1$ then $weight((v, w)) \in [0, 1]$. Figure 1 shows that 93% of the reblogs occur during the first day and additionally, at Figure 2 it is shown that actually 82% of the reblogs occur during the first 7

[4] *Application Programming Interface*, it is a set of routines, protocols, and tools for building software applications.

hours (it is summarized in Table 1). Hence, we have chose $\tau = 7$ *hours*. This results are similar to those found in Twitter [11]. Thus, we assign the weights to each network arc in T_1 as the reblog rate for each post which was posted in $[t_{ev} - 2$ *weeks*$, t_{ev} - 7$ *hours*$)$. We show the obtained weight distribution in Figure 3, where we can observe a non-uniform distribution.

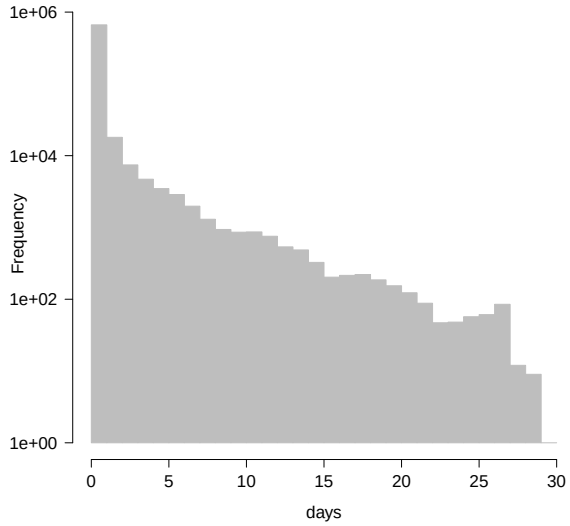

Fig. 1: Reblog delay in days (y log-scale)

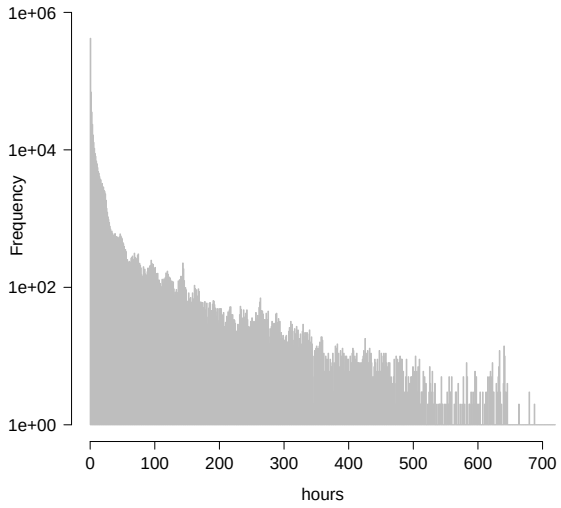

Fig. 2: Reblog delay in hours (y log-scale)

Days	Cumulative	Hours	Cumulative
1	93%	1	56%
2	96%	2	69%
3	97%	3	73%
4	97%	4	77%
5	98%	5	79%
6	98%	6	81%
7	99%	7	82%

Table 1: Cumulative reblog percentages.

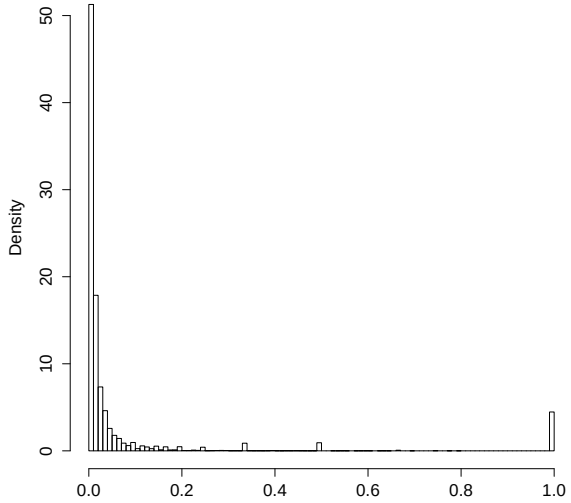

Fig. 3: Weight distribution for $\tau = 7$ *hours*

3.3 Obtaining the Ground Truth Ranking

We are going to consider the *future* diffusion data for setting our *ground truth* as follows. For each vertex v, it is assigned the average cascade size of each post (not re-post) it published during $[t_{ev}, t_{ev} + 2$ *weeks* $- 7$ *hours*$)$. Once we have computed the average cascade size for each vertex, they are sorted by this number yielding a ranking. Note that, in the same way, we can

compute the ranking in T_1 and it could be considered as a metric. Actually, we have considered the T_1 ranking as a reference metric, our benchmark. Since every metric will use the weighted graph with weights computed using T_1 diffusion data, we could think this T_1 ranking should not be outperformed. Figure 4 shows the T_1 ranking performance over the T_2 ranking. We are using a measure called *recall* which is widely used in information retrieval. We are going to define it as follows.

$$\text{top-}m \textit{ recall} = \frac{|\text{top-}m \text{ nodes}(\textit{refRank}) \bigcap \text{top-}m \text{ nodes}(f)|}{m}, \tag{9}$$

where *refRank* is the reference ranking (the T_2 ranking) and f is the rank to evaluate (the T_1 ranking). This measure is a rate defined as the first m common elements in both rankings over the maximum common possible (m). Then, in Figure 4 we can observe the recall of the ranking T_1 when predicting the ranking T_2. Note that the top-1 ranking has 0% recall (first dot), the top-5 has 40% and from the top-10 onwards the recall goes between 40% and 60% while increasing the top users to be predicted. We are not showing further from top 50% because it does not provide additional relevant information.

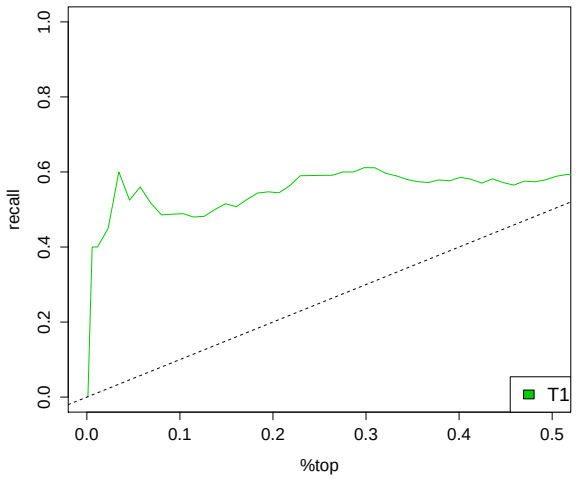

Fig. 4: T_1 ranking performance.

As shown in Figure 5 there is a peak of activity (just after t_{ev}, which is vertically dotted) which might bias the results of the T_2 ranking (ground truth). In order to check the relevance of this activity peak we compute the ranking for T (whole period) and use T_1 and T_2 rankings (recall that $T_1 \subset T$ and $T_2 \subset T$) for obtaining how much information each ranking gives (see Figure 6). Since we see that the T_2 ranking outperforms the T_1 ranking, we might think the information peak after t_{ev} which is used in the T_2 ranking could be adding noise for the T_1 ranking. In oder to check this phenomenon, we take the T_1 period, and split it in T_1' and T_2' periods (of same length). Again, we compute the rankings for this 2 periods and compare them with the T_1 ranking. We do it for different length periods, and also for the T_2 period but avoiding the activity peak. The results are very similar to the first one in Figure 6 that compares T_1 and T_2 on T (see an example in Figure 7). This seems to imply that about half of the most influential users are oscillating over time, as in the case of the 2^{nd} most influential user in T_2, had just 1 post during T_1 and no re-blogs. This fact agrees with [7] where is observed that maintaining influence requires personal effort. Hence, the observed activity peak could be just highlighting this phenomenon.

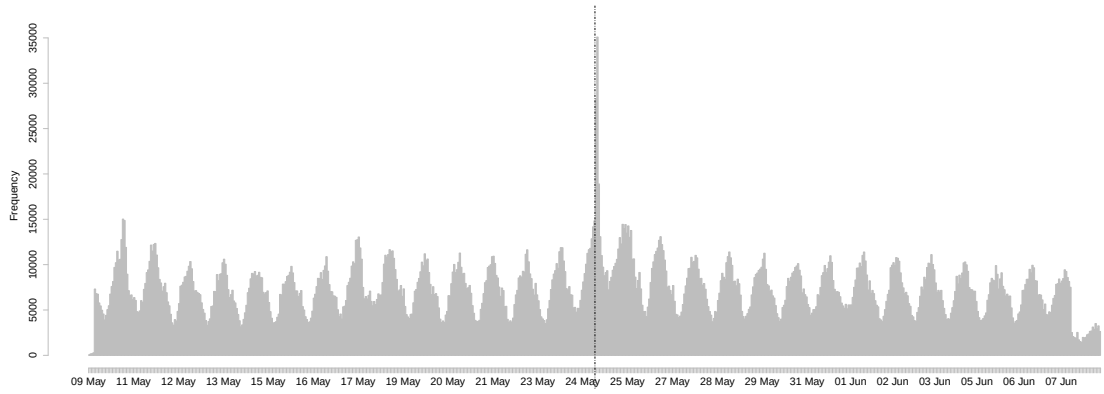

Fig. 5: Activity histogram

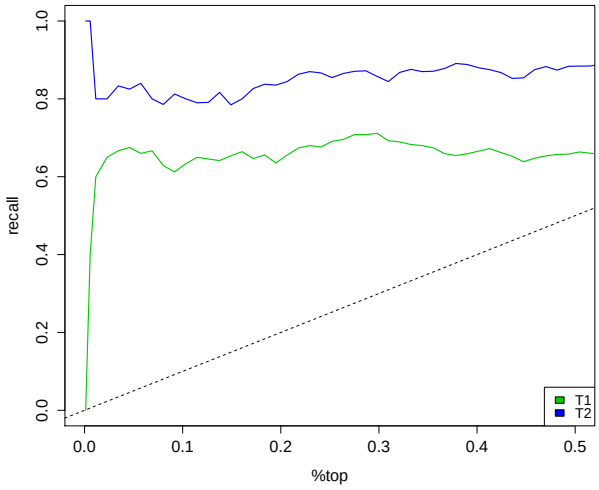

Fig. 6: T_1 and T_2 ranking performance on T.

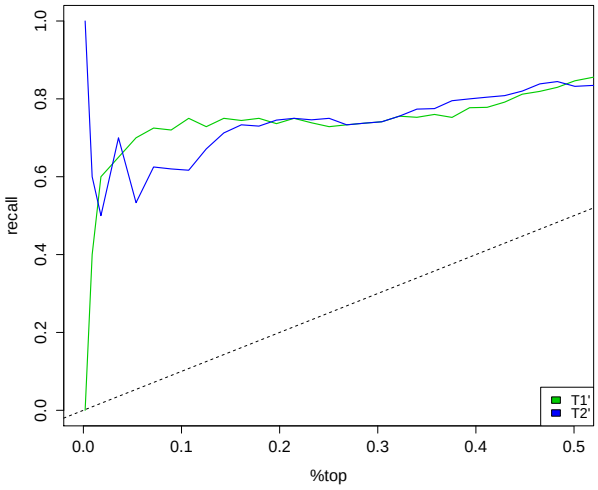

Fig. 7: T_1' and T_2' ranking performance on T_1.

4 Performance Evaluation

The ranking error for each proposed metric is shown in this section. Global and partial ranking errors are important to determine the performance of a ranking. We have already tested those metrics which have a *weighted* version with their *unweighted* versions in order to check the relevance of adding the diffusion data (weights) to the network. The result is that the *weighted* versions outperform those which are *unweighted* (critically in some cases). Hence, we are just showing the *weighted* versions of this metrics.

4.1 Global Ranking Performance

For measuring the global ranking we define the following function.

$$absErr = \frac{\sum_{i=1}^{N} |\text{refRank}_i - pos(f(\text{refRank}_i))|}{N}, \tag{10}$$

where *refRank* is the reference ranking (the T_2 ranking), f is the metric function, and *pos* gives the position of the ranked vertex. Figure 8 shows the distribution of each summatory term of

absErr for each metric.

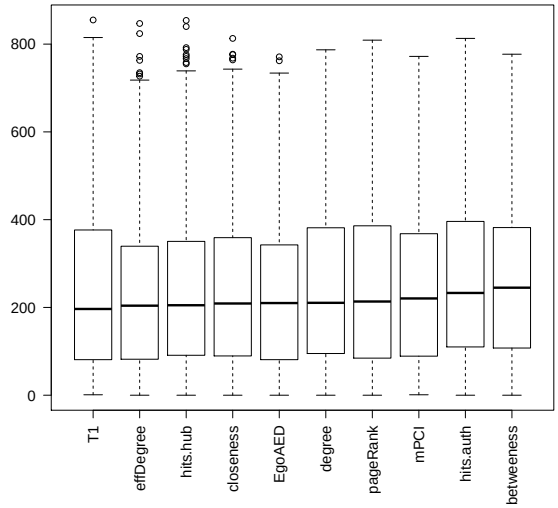

Fig. 8: Global Absolute Error for each ranking.

Metric	Spearman's ρ	*absErr*
Degree	0.1324	248.82
Betweenness	0.1346	266.42
Closeness	0.0935	245.43
μ-PCI	0.1104	246.32
PageRank	0.0935	251.03
HITS.auth	0.1004	267.33
HITS.hub	**0.1584**	**240.87**
Effective Degree	0.1332	**236.10**
EgoAED	**0.1515**	**234.84**
T_1	0.0891	247.89

Table 2: Ranking's correlation and error.

As can be seen in Figure 8, there is no big difference among the metrics considering this error measure. Moreover, we show in Table 2 both the Spearman's rank correlation coefficient (ρ) and *absErr* for each metric, using the T_2 ranking as the reference one. On the one hand, we have highlighted the metrics with higher ρ, which means a high ranking correlation. On the other hand, we highlight the metrics with lower absolute ranking error. However, it is observable that all metrics have similar performance for the global ranking, with no big differences among them.

4.2 Partial Ranking Performance

The next step is considering the top ranking and study the evolution of such ranking when we increment the fraction of top vertices considered. For this purpose, we are going to use another *goodness of fit*, the top-*recall* as defined in Section 3.3. This way, the top-10 recall, for instance, is the amount of nodes in the top 10 metric ranking which are in the actual top 10 reference ranking. For this work, the recall plots are going to show the top-1, top-5, top-10, top-20, top-30, and so on.

At each plot, we also plot the curve of the T_1 ranking, which is considered as a reference (it must be the same as it is in Figure 4).

First, we consider those metrics described in Section 2.1 and those proposed in this work because of their similarity with the degree metric. The results are shown in Figure 9, where we can notice several features. In general, betweenness and closeness are outperformed by the other metrics, even when closeness performance from 10% on is quite competitive (the first 10% of the nodes is approximately the top-80). Also, we can check that μ-PCI slightly outperforms the degree ranking, which is consistent with what we expected. Finally, the effective degree, EgoAED and the T_1 ranking are close each other and clearly outperform the other metrics.

Rankings described in Section 2.2 are shown in Figure 10. Surprisingly, PageRank is able to recall succesfully the top-1 ranking, what might be considered lucky if it had not a good performance, in general. Among the HITS metrics, the hub score clearly outperforms the authority score. However, both HITS rankings have a low performance at the first 20% top ranking.

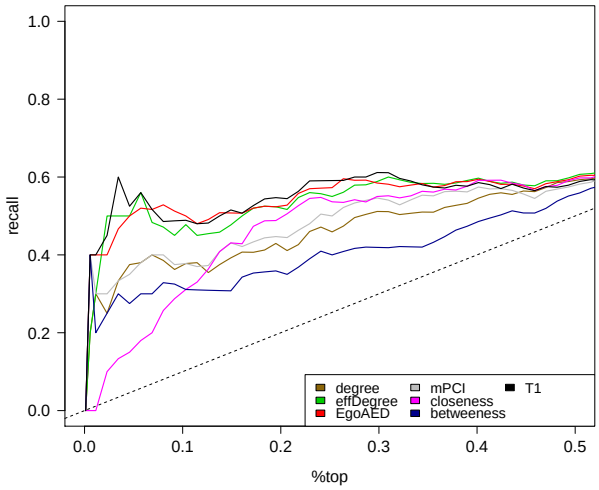

Fig. 9: Ranking recall for centrality metrics.

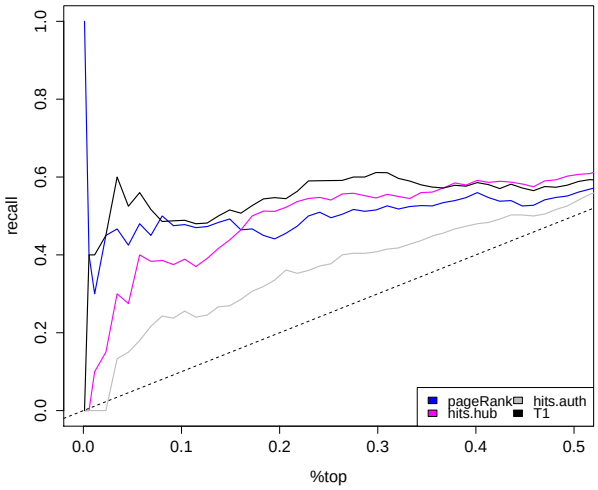

Fig. 10: Ranking recall for other centrality metrics.

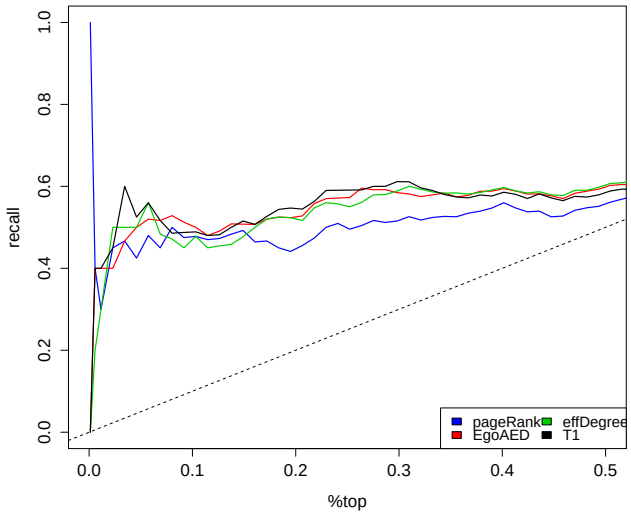

Fig. 11: Summary of the metrics with higher ranking recall.

In order to show and compare the metrics with higher performances we have summarized them in Figure 11.

5 Conclusions and Future Work

This paper analyzed an actual diffusion dataset from an OSN called *Tumblr*, from where we have built a weighted network based on the relationship activities. We aimed at ranking the nodes in the network in order to retrieve the most relevant ones. For this purpose, we took the last (temporal) part of the diffusion data for being the *ground truth* (or "solution"). We proposed some ranking metrics for our aim, which also were easy computable. These metrics have been tested and compared with a range of centrality metrics widely used in the literature, where, in general, our proposed metrics outperformed the other ones. Besides, the diffusion data itself (T_1 ranking) has behaved as one of the best rankings in our tests. However, we cannot say that any

of the techniques has a high performance, aside from PageRank ranking with the top-1 recall.

Also, we have checked some previous results previously observed on Twitter such as a low reciprocity rate [7], or similar reblog/reshare/retweet delay and rate [11]. Besides, we might have experimented oscilations in the influential ranking due to the occurrence of an event (t_{ev}), as claimed in [7].

Regarding the current difussion models for epidemics or rumors, we have seen that they do not match real diffusion data due to the fact that they usually have a uniform propagation success rate (λ or β), which might seem to be far away from reallity. However, thay have good results [12, 18]. So, we are really wondering if these models could be improved somehow, being closer to reallity, or their approximations are quite good enough.

Finally, we should test our experiments in other OSN's as well as using larger networks, in order to check if we have similar results.

References

[1] Roy M Anderson and Robert McCredie May. *Infectious diseases of humans*, volume 1. Oxford university press Oxford, 1991.

[2] Eytan Bakshy, Jake M Hofman, Winter A Mason, and Duncan J Watts. Identifying influencers on twitter. In *Fourth ACM International Conference on Web Seach and Data Mining (WSDM)*, 2011.

[3] Alain Barrat, Marc Barthelemy, Romualdo Pastor-Satorras, and Alessandro Vespignani. The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(11):3747–3752, 2004.

[4] Pavlos Basaras, Dimitrios Katsaros, and Leandros Tassiulas. Detecting influential spreaders in complex, dynamic networks. *Computer*, 46(4):24–29, 2013.

[5] Murray A. Beauchamp. An improved index of centrality. *Systems Research and Behavioral Science*, 10:161–163, 1965.

[6] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International Conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[7] Meeyoung Cha, Hamed Haddadi, Fabricio Benevenuto, and P Krishna Gummadi. Measuring user influence in twitter: The million follower fallacy. *ICWSM*, 10(10-17):30, 2010.

[8] Yi Chang, Lei Tang, Yoshiyuki Inagaki, and Yan Liu. What is tumblr: A statistical overview and comparison. *SIGKDD Explor. Newsl.*, 16(1):21–29, September 2014.

[9] Guilherme Ferraz de Arruda, André Luiz Barbieri, Pablo Martín Rodríguez, Francisco A. Rodrigues, Yamir Moreno, and Luciano da Fontoura Costa. Role of centrality for the identification of influential spreaders in complex networks. *Phys. Rev. E*, 90:032812, Sep 2014.

[10] Manuel Garcia-Herranz, Esteban Moro, Manuel Cebrian, Nicholas A. Christakis, and James H. Fowler. Using friends as sensors to detect global-scale contagious outbreaks. *PLoS ONE*, 9(4):e92413, 04 2014.

[11] Sysomos Inc. Replies and retweets on twitter. `http://www.sysomos.com/insidetwitter/engagement/`, 2010.

[12] Fang Jin, Edward Dougherty, Parang Saraf, Yang Cao, and Naren Ramakrishnan. Epidemiological modeling of news and rumors on twitter. In *Proceedings of the 7th Workshop on Social Network Mining and Analysis*, SNAKDD '13, pages 8:1–8:9, New York, NY, USA, 2013. ACM.

[13] Elihu Katz and Paul Felix Lazarsfeld. *Personal Influence, The part played by people in the flow of mass communications.* Transaction Publishers, 1970.

[14] Maksim Kitsak, Lazaros K Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Eugene Stanley, and Hernán A Makse. Identification of influential spreaders in complex networks. *Nature Physics*, 6(11):888–893, 2010.

[15] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, September 1999.

[16] Mark Newman. *Networks: An Introduction.* Oxford University Press, Inc., New York, NY, USA, 2010.

[17] Mark EJ Newman. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Physical review E*, 64(1):016132, 2001.

[18] Donal Simmie, Maria Grazia Vigliotti, and Chris Hankin. Ranking twitter influence by combining network centrality and influence observables in an evolutionary model. *Journal of Complex Networks*, 2(4):495–517, 2014.

[19] Rudra M. Tripathy, Amitabha Bagchi, and Sameep Mehta. A study of rumor control strategies on social networks. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1817–1820, New York, NY, USA, 2010. ACM.

[20] Shinji Umeyama. An eigendecomposition approach to weighted graph matching problems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 10(5):695–703, 1988.

[21] Daijun Wei, Xinyang Deng, Xiaoge Zhang, Yong Deng, and Sankaran Mahadevan. Identifying influential nodes in weighted networks based on evidence theory. *Physica A: Statistical Mechanics and its Applications*, 392(10):2564 – 2575, 2013.