User profiling on Twitter

Edgar Rocha [a] Alexandre P. Francisco [a] Pável Calado [a] H. Sofia-Pinto [a]
[a] *INESC-ID / CSE Dept, IST, Tech Univ of Lisbon, Portugal*

Abstract. Social networking and microblogging integrating services, such as Twitter, have been gaining popularity in recent years. In this context, the study of user activity and information flow raises several interesting questions, with important real life implications, such as user influence prediction and information flow optimization. In this paper we study how to differentiate users given their activity. We focus just on user activity, ignoring the content of messages a user exchanged. Unlike previous work that focus on user activity and content of messages user exchanged, we take into consideration both social interactions and tweeting patterns, which allow us to profile users according to their activity patterns.

Keywords: User interactions, activity pattern mining, social networks, Twitter

1. Introduction and Motivation

Social networking and microblogging services have been gaining popularity in recent years. These services reflect social relations among people, such as user communities and common interest groups, and allow users to share content, through typically short, but informative, text messages, which may include links to images, videos or Web pages. Since the number of users in these systems can go up to planet scale, the inherent information flow is of extreme relevance to understand their interests, behaviour and also to discover near real time events and news, most of the time even before news media take notice of them.

In particular, we observe that users behave differently. Understanding users and being able to characterize them, for instance, as influential or not, has a high market value in areas such as marketing strategy or trends prediction. However, it is not a trivial problem, since many issues contribute to what actually defines a user:

- A simple approach is to analyse the **network structure** resulting from users and their connections with other users, e.g. by analysing for each user the number of friends or the number of followers [13].
- However, this may be insufficient. For instance, a leading scientist in Physics can hardly compete with an Hollywood celebrity, if one only takes into account the number of followers. This occurs because the underlying structure of groups of users can have different shapes and scales. Therefore, it is also important to examine users' interests, gender, sex, age, geo-localization, and other characteristics of the **user profile** [11], since these can provide some kind of social context.
- Research has also shown that **post content**, i.e. the actual messages posted by the users, is also important. To this effect, simple tools, such as examining the message keywords, or more complex tools, such as sentiment analysis, can also play some role on user characterization [10]. This is a very active area of research nowadays.
- Moreover, it is observed in daily life that even not very knowledgeable and or initially very highly regarded people can become influential, if they have good social skills. These people often spread their ideas in a fervorous way, almost as in an evangelism campaign. In this case, **user activity**, i.e. the number of posts and/or the number of answers can better reflect the true value of the user in terms of influence upon others [2].
- Finally, taking it one step further, one can try to analyse the consequences of the user's interactions upon others. In this case, one can observe, for instance, if other users respond to the user's posts, how many times they do so, if the ideas that are being proposed are accepted by others, i.e. if

they get spread by others in return from the user activity. In this case, one takes into account the **dynamic patterns** arising from user activity [4].

In this work, we try to address the following question: how do we differentiate users, from a behavioural point of view? To do so, we combine information extracted from network structure, user activity and the users' dynamic patterns features. Answers to this question will allow for more accurate user profiling, with practical implications for user influence prediction, trends prediction and efficient information flow and marketing strategies.

Our approach relies on data extracted from Twitter. Using a dataset of about 1,350,000 users, together with their underlying network of friends and followers, and about 3,870,000 posts, we address several issues. We want to answer questions, such as: What are the differences in twitting behaviour between different types of users (e.g. regular users versus evangelists)? What is the impact of a user's post on the remaining users? What are the paths taken by the user's post in the information propagation chain? What is the position of a user in this propagation chain? Is this tweet spreading as a star or tree (for instance, celebrity's posts tend to spread in star propagation chains, while normal user's posts tend to spread in tree propagation chains)? This was achieved by analysing several network and information flow measures, such as tweet nature and user participation delay, which allowed us to classify users with high confidence into six different classes, ranging from high active users and content makers through normal users that mostly just follow content.

This paper is organized as follows. Section 2 covers relevant related work organized according to the used approach. Section 3 describes the methods we used to extract data from Twitter and discusses the algorithms proposed to infer retweet chains and network chains. We also describe the cluster method and the features used. Section 4 discusses our results. In Section 5 we conclude and discuss possible interesting future work.

2. Related work

Analysing users and their behaviour on online social networks has been the subject of many previous works [9,7,8]. The particular domain of the Twitter microblogging service has not been an exception. By looking at the contents produced by users, or at the actions they perform, researchers have been able to derive user characterizations and other useful information, with the goal of, for example, doing sentiment analysis [10] or predicting the diffusion of information [14].

An example is the work of Chu *et al.* [3], where they observe the differences between human users and what they designate as *bot* and *cyborg* users. The authors characterize a bot as a user whose actions are all automatic, i.e. without any human intervention. A cyborg is similar to a bot, but performing some human-assisted actions, such as providing RSS feeds mixed with the user's comments, thus showing characteristics of both humans and bots.

By performing data analysis on tweeting behaviour, tweet content and account properties, the authors observe that bots generate fewer tweets than humans, whereas cyborgs have the highest tweet count, due to the combination of both human and bot behaviours. Another interesting result is the fact that humans tend to have a number of friends close to the number of followers, which shows a reciprocal behaviour. Bots and cyborgs also attempt to have a similar friend/follower ratio. However, this is accomplished artificially, by cancelling friends that are not followers, thus mimicking human behaviour.

When examining timing properties, the authors show that bots maintain the same activity level every day of the week, with a slight drop late during the night. However, in the case of humans, the pattern is similar but less activity is shown during the weekend. Using the entropy rate of tweet intervals, Chu *et al.* concluded that humans have a high entropy level, due to their complex timing behaviour, while bots and cyborgs have a regular timing behaviour, which results in low entropy.

In a similar manner, Java *et al.* propose a taxonomy of user intentions on Twitter [6]. To achieve this, users were manually categorized according to their link structure and tweet contents. Based on link structure, three main categories of users where found: (1) Information Sources—a user that can be seen as a hub and has a large number of followers; (2) Friends—where most of users belong to, forming social networks of friends, family, co-workers, among others; and (3) Information Seekers—users that post very few times, but follow other users, thus regarding Twitter mostly as a source of information.

Based on content, user intentions were classified into four categories: (1) Daily Chatter—the most common use of Twitter, by users who mainly post about their daily routine; (2) Conversations—where dialogues are established by posting and replying to posts; (3) Sharing Information—when users intend to share information by posting an URL; and (4) Reporting news—where we have users that report the latest news or comment about recent events.

Tweeting behaviour, network structure, and the linguistic content were used by Pennacchiotti *et al.* to infer the political orientation and ethnicity of users [11]. They show that network features perform well when used to classify user political orientation. According to the authors, this occurs due to the interaction between users and media or personalities with an established Twitter presence. On the other hand, linguistic features are useful to determine ethnicity, since African-American users tend to use a specific set of words.

While still trying to classify Twitter users, other researchers have focused on a different set of characteristics, namely, the user's ability to influence others or to divulge information. Cha *et al.*, for instance, define three types of influence on Twitter: Indegree, Retweet and Mention [2]. *Indegree influence* regards the user's popularity, and is measured by the number of followers. *Retweet influence* regards the tendency of the user's audience to retweet her posts, and is measured as the total number of retweets. *Mention influence* regards the likelihood that the user will be mentioned in other users' posts, and is measured by the number of times the *@username* tag occurs within tweets.

Interestingly, the authors observed that most popular users are not necessarily the most mentioned, nor the most influential. In fact, influence is gained from an effort in personal involvement between a user and its audience. Among the most influential users, it was observed that mainstream news organizations are the most retweeted, although they are not the most mentioned. On the other hand, celebrity users (e.g. famous artists) are more often mentioned than retweeted. Although less influential, the authors also refer to *local opinion leaders* and *evangelists*, as users who, besides being often retweeted, are also able to maintain an active dialogue with their audience.

In a very recent work, Petrovic *et al.* [12] conducted an experiment showing that humans can predict which tweets will be retweeted, just by looking at tweet content. These results were then compared to those of an automatic classifier, which uses social features and content features to automatically predict retweets.

Social features include the number of followers, friends, user status, favourite tweets, number of times the user appears in other users' lists, if the user's language is English and whether the user is verified. If the identity of a user is susceptible of being confused with other users (e.g. Obama) then it needs to get verified. Once verified its authenticity is guaranteed by Twitter. Content features include the number of mentions, URLs, occurrence of trending words, tweet length, novelty, whether the tweet is a reply, the actual words used and the number of hashtags. Hashtags are words that can be used to follow specific topics, and are indicated by a starting hash character, e.g. *#Portugal*.

Network Structure	**User Profile**	**Post Contents**	**User Activity**	**Dynamic Patterns**
[2] [3] [6] [8] [9] [13]	[9] **[11]**	**[2] [3] [6]** **[10] [11]** **[12] [14]** [7] [8]	**[2] [11]**	[4] [5]

Table 1

Previous work, organized according to the features explored.

The features that proved to have better results in predicting retweets were the number of followers and presence in user lists. The authors show that tweets written by verified users have a higher probability of being retweeted. This is the case of most celebrities, who thus have more ability to diffuse content and cause more retweet chains.

All work described so far concerns Twitter. Gomez-Rodriguez *et al.* [5] developed a method to trace diffusion and influence paths through the network on a dataset of MemeTracker. In this work authors try to infer the network based on recurrent patterns of diffusion between different nodes, i.e. if node A and B always have a similar text with different timestamps, then there is a possible edge between node A and B. Moreover, authors do not observe the content of diffused posts, but they cluster a set of phrases to aggregate different phrase variants instead. Therefore, for the set of posts in the same cluster and looking at their timestamps, the authors infer the chains or paths of diffusion.

Since users do not influence all their neighbours in the same way, it is possible to use this methodology to infer the real patterns of interaction between users and their neighbours, i.e. instead of doing an analysis based on user network connections, one should focus on the result of the interactions between a user and the surrounding neighbourhood to study user behaviour.

To the best of our knowledge, none of previous research work on user profiling in Twitter explores dynamic pattern features. Table 1 summarizes previous related research on user and social network analysis. References are organized according to the set of features explored. In boldface we show work that focuses on Twitter. Our proposal, also aims at characterizing Twitter users. However, and unlike any previous work, we combine network structure, and user activity with dynamic patterns features. Thus, we are able to map user produced content into the networks of followers. This reveals how information progresses through the network, and how users react to such information, yielding patterns of diffusion that can then be applied in profiling user behaviour. We detail our approach in the following sections.

3. Method

To better understand users based on their dynamic behaviour we decided to use Twitter data, because the provided API gives us easy access to content and network data. Our approach relies on data collected from Twitter and on the Expectation-maximization (EM) based clustering method. In this section we detail how data was acquired, how features were extracted by content analysis and network mining, and how the clustering method was applied to achieve a user profile classification.

3.1. Data acquisition

Twitter highlights particular words as trends. These can be defined either by users or by Twitter's Trending Topics algorithm. On the 8th November 2010, we started searching 30 topics and trends (see first 30 rows in Table 2). On the 18th of the same month we added `crisis`, `economy`, `java`. All data acquisition was stopped on the 10th of February. Each topic was extracted through the Twitter Search API[1], where content is provided as RSS feeds. The result for each topic was an RSS feed with a set of entries, where each entry contains the text of a tweet, its author, publication date and a unique identifier. When there are no new tweets between calls to RSS feed, the last tweets are returned. Thus, to avoid duplicated tweets, one needs to store all unique IDs that were previously collected and check every new entry against the IDs list. We used the Twitter Search API because the communication protocol is easy to implement, multiple IPs can be used and the order of words in queries is preserved. The order of words in queries is important because we are searching for specific names of movies and events, thus we want to find tweets that have the full name specified in the query, instead of tweets that simply have one or more query words.

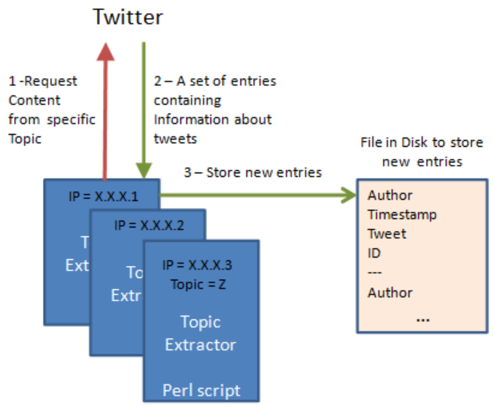

Fig. 1. Information flow for content extraction for each topic. All information extracted is stored in files and each script has associated a different IP.

Requests to the Twitter Search API were performed with an interval of 5 minutes, due to Twitter limitations on request rates. Since we had several topics, we associated an IP address to each topic (using a single machine with several configured IPs) and, also for each topic, we ran a Perl script for querying the Twitter API, parsing results and storing tweets in one file per topic, as depicted in Fig. 1.

3.2. Content analysis

Twitter allows users to post tweets in any language, using non-Latin characters. Twitter tries also to ban users who spam or post with the main purpose of spreading malware, virus or aggressive advertisement. Since we want to ensure that we can analyse Twitter contents and we want to avoid tweets that are spam, we needed to filter tweets posted by banned users/authors and tweets that had non-Latin characters. Thus, each tweet that contains a banned user was excluded from the list of tweets to be analysed. The same happens for tweets that have more than 10% of non-Latin characters, i.e. for a tweet that uses the maximum of 140 characters, it can not contain more than 14 non-Latin characters.

As mentioned above, in this work we are interested on information flow and user activity patterns and, thus, it is important to track both content and user ac-

[1] http://search.twitter.com/api/

Table 2
The 33 threads of Twitter data.

Topic Name	# Tweets	Average Tweets/Day	# Retweets	Average Retweets/Day
#OE2011	15	0.2	0	0
#christmas	284037	4177.0	28881	42.4
#xbox	89509	1316.3	3704	54.5
#ipad	658765	9687.7	81723	1201.8
#kindle	91160	1340.6	6725	98.9
#wikileaks	250245	3680.1	138684	2039.5
#f1	113642	1671.2	19482	286.5
#london2012	2609	38.4	1158	17.0
#atp	6925	101.8	923	13.6
#pga	13356	196.4	463	6.8
#dakar	5102	75.0	677	9.9
#euro2012	261	3.8	16	0.2
Red Bull Air Race	258	3.8	4	0.1
pirates of the caribbean 4	7433	109.3	1489	21.9
kung fu panda 2	4721	69.4	228	3.4
hangover 2	28226	415.1	3628	53.4
transformers 3	21568	317.2	2628	38.6
sherlock holmes 2	5167	76.0	858	12.6
twilight breaking dawn	33659	495.0	4909	72.2
mission impossible 4	2003	29.5	255	3.8
harry potter	744898	10954.4	56665	833.3
#avatar	13771	202.5	688	10.1
#flu	15204	223.6	1025	15.1
#h5n1	647	9.5	35	0.5
#surf	13153	193.4	1176	17.3
#nobel	3277	48.2	1060	15.6
#economical	280	4.1	16	0.2
#israel	107471	1580.5	24722	363.6
#iraq	24938	366.7	6472	95.2
#terrorism	16243	238.9	2021	29.7
crisis	463507	7022.8	61034	924.8
economy	599693	9086.3	72693	1101.4
java	253905	3847.1	24857	376.6

tions on Twitter. The most direct way for tracking content on Twitter is to identify tweets and retweets, the two types of posts that a user can do. A tweet is a post done by a person. A tweet does not need to be related with previous posts (either from the same author or from another user). A retweet is a copy of a tweet that a user considered interesting enough to diffuse to his followers, by posting an exact, or slightly different, copy of the original tweet. Although both types of tweets can be done via the Twitter web interface, there are users who do not use the proper web interface button to retweet, preferring manual text manipulation to do their own retweets, or they rely on external applications. Therefore, since those users use text manipulation, they can post retweets using different tags instead of the normal *RT* tag. In this work, following the work done by Boyd *et al.* [1], we identified retweets by performing case insensitive match with the following patterns:

- *rt @username ...*;
- *retweet @username ...*;
- *reading @username ...*;
- *retweeting @username ...*;
- *... from @username*;
- *... rt @username*;
- *... via @username*;

where *username* identifies the author of the original tweet. Note that a user can use two or more retweet forms at the same time, e.g. *RT @username1 Osama is dead. via @username2*, and in this case the retweet tag that appears in beginning of sentence takes a higher priority and *username1* is chosen as author of the original tweet, ignoring *username2*. Moreover, we can identify another kind of interaction: a reply. A reply is a particular case of a tweet and is the result of an explicit interaction between two users, i.e. a reply is a message directed to another user. A reply begins with the username or screen ID of the message recipient, e.g. *@username How are you?*.

In the context of this work, to analyse user activity, we computed for each user/author the number of posts (tweets and retweets), tweets, retweets and replies.

3.3. Retweet chains

Tweets and retweets provide important information to reconstruct information diffusion processes on Twitter, since retweets are the direct result of the diffusion of tweets. Also, when a user posts a retweet there is an implicit evaluation of its relevance and interestingness. Thus, through the analysis of these processes, it is possible to understand user interests, how information diffuses, who are the users that diffuse more information, and who are the users that have more capacity to diffuse or to influence people. In this section, we describe how to identify and reconstruct these diffusion processes or retweet chains. Later we discuss how these chains relate with the underlying Twitter social network and how network data can improve chain identification.

To extract retweet chains, one needs to have a set of tweets and a set of retweets including tweet/retweet text, timestamps, authors of tweets and, in case of a retweet, both original tweet and retweet author. The algorithm to reconstruct retweet chains is as follows:

1. For each retweet R1, find in the set of tweets the tweet T1 that: (a) contains a timestamp less than the R1 timestamp; (b) contains the same text; and (c) where the author of T1 is the same as the tweet author specified in R1;
2. For each retweet R1 find in the set of retweets all retweets that have equal text;
3. For each retweet R2 in the previous set, find retweets R3 for which (1) R2 timestamp is lower than R3 timestamp and (2) R3 tweet author is the same as R2 retweet author.

Note that, in step 1, we identify the chain source, i.e. the original tweet that users retweeted originating a chain of retweets. In step 3 we try to detect retweets of retweets, i.e. given that users can introduce changes on retweets, we can have for instance *RT @username2 RT @username1 Portugal is great!* where *username2* retweets a retweet by *username1*.

From the inference of retweet chains it is possible to extract some measures that can be used to characterize diffusion processes, topics extracted and users involved. These measures are: the number of retweet chains by topic, which shows the tendency of a topic to create chains; the number of user participations in retweet chains, which shows how a user tends to join chains; the number of user participations in retweet chains where the user is the author of the first tweet, i.e. the root node, which measures the ability of a user to post interesting content or to diffuse content; the length of retweet chains, which measures the ability of a specific tweet/topic to diffuse on the network; and the position of users in retweet chains, that provides a measure of user reaction.

3.4. Network analysis

The Twitter API[2] allows us also to extract the list of followers and friends for a given user. A friend is a user that is followed by the current user, and a follower is a user that follows and receives status updates from the current user. Responses through the API can be either in JavaScript Object Notation (JSON) or Extensible Markup Language (XML). Every user in Twitter has a screen name, e.g. barackobama, and has a user identifier, e.g. 92869723. The API supports two different requests to get the list of followers for a given user:

http://api.twitter.com/1/
*followers/ids.**format**?user_id=**X***
http://api.twitter.com/1/
*followers/ids.**format**?screen_name=**X***;

where the keyword *format* specifies the format of the output, either XML or JSON, and *X* must be replaced by the numeric identifier in the first case and by the screen name in the second case. There are also two different requests to get the list of friends for a given user:

http://api.twitter.com/1/
*friends/ids.**format**?user_id=**X***
http://api.twitter.com/1/
*friends/ids.**format**?screen_name=**X***

where parameters are defined as above.

The results of these calls may be split through several XML pages. For users with a large number of followers and/or friends, the results must be explicitly split, otherwise Twitter API returns an HTTP exception, known as *Fail Whale*, with code 502. Twitter only returns a maximum of 5000 users per XML page and, for that reason, a user may require more than one call to be totally explored. To use previous requests with XML pages, all one needs to do is to add the string "*&cursor=-1*" to the end of each HTTP request. Also, if some user is marked by Twitter as a spammer or if it does not respect the rules of Twitter, it is possible that it gets banned and, in that case, when its followers or friends are requested an HTTP exception with value 400 is returned.

These calls to Twitter API do not require authentication if user profiles are public, otherwise OAuth authentication is required. Private profiles in Twitter can only be viewed by authorized users. Fortunately most Twitter profiles are public. We note also that all the other calls specified in the Twitter API require authentication, even if the profiles are public. For the previous API calls, when authentication is required, if the friends or followers of a user are requested with an anonymous call, an HTTP exception is returned with code 401.

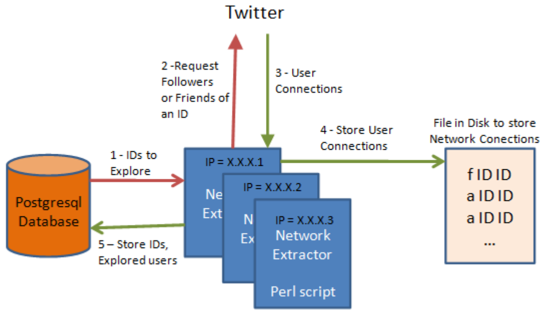

Fig. 2. Network extraction. All IDs are stored in a single database.

Another important issue is the request limit. Twitter imposes limits on the number of calls per hour. In the case of anonymous calls, the limit is 150 requests/hour per IP. In the case of authenticated requests the limit is 350 request/hour per user. Authenticated requests can be used but they require that each user accepts the crawler application and gives rights to extract requested information, even if profiles are public. Since we want to extract the network for a large number of users, we need to increase the limit of anonymous calls. Thus, we used 30 different IP addresses. More IP addresses could be used, but that would introduce an overhead information storage given the data volume and the number of concurrent processes. Note that in our solution all new users were stored in a centralized database, as we can see in Figure 2. Since Twitter has many servers to answer API requests, when we request the current limit for an IP address, it is important to consider that the server can be outdated and, to avoid requests over the limit, we stop at 147 requests/hour per IP address. We can get the current limit with the request:

http://api.twitter.com/1/
*account/rate_limit_status.**format***;

where the keyword *format* can be replaced by XML or JSON and the returned limit is related to the IP address that made the call.

In our solution we assign a file to each process/script where each network connection extracted from Twitter is stored either as *a 1 2* or *f 1 2*, where *a* means a friend connection and *f* means a follower connection. In other words, for connection *a 1 2*, *user 1* has as friend *user 2* and for connection *f 1 2*, *user 1* has as follower *user*

[2] http://dev.twitter.com/

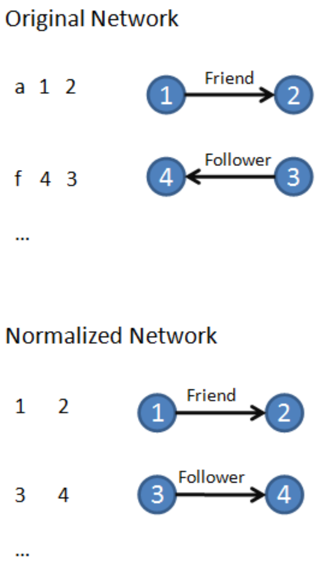

Fig. 3. To normalize the stored network one needs to put all connections/edges in the same direction.

2. It is important to distinguish these two type of connections because of the direction of information flow in the network. In a friend connection, *user 1* receives information from *user 2*. However, in a follower connection *user 1* diffuses information to *user 2*.

Since the network can be huge, we may not be able to load all data directly into RAM memory to be analysed. For this reason we use Webgraph[3], a framework for graph compression aimed at studying large Web graphs. It provides a simple way to manage very large graphs, exploiting modern compression techniques. Webgraph provides also a set of tools to iterate over and process huge graphs, either online or offline. When offline, the graph is processed but not loaded into memory.

Webgraph has a specific input format to be able to compress and load a graph. Since the network is stored in format *a|f ID ID*, one needs to normalize all edge directions. More specifically, as we can see in Fig. 3, for follower connections we have, for instance, *f 1 2*, which means that *user 2* is connected to *user 1* and, for connection *a 3 4*, *user 3* is connected to *user 4*. So, one needs to invert all follower connections to be able to build the graph without the need to look at the connection type.

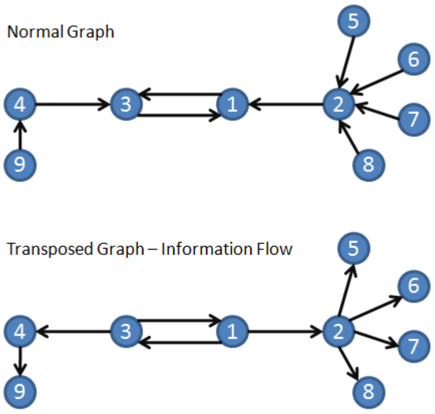

Fig. 4. To analyse information flow one needs to transpose the normal graph.

It is important to take into account that the normal graph shows us how users are connected and who is a friend or follower of whom, but if we want to analyse the information flow through the network we have to transpose the graph. For instance, if we consider that *user 4* is connected to *user 3* on Twitter, then *user 3* sends information to *user 4*, as we can see in Fig. 4. The graph may be transposed and compressed offline also using Webgraph tools.

In our work we started to extract user network on the 20th of February and stopped on the 10th of June. The crawling of users was started at nodes that participated in tweets previously extracted. Then, all neighbours and subsequent nodes were crawled in a breath first search mode. In the network extracted, 129982336 nodes and 2528951571 edges were collected. The uncompressed graph takes 55GBytes of memory. When compressed with Webgraph it only takes 3.5GBytes.

3.5. *Network chains*

Although we have already identified retweet chains, to understand information diffusion processes and user activity on the Twitter network, it is important to correlate retweet chains with network structure. In this section we propose a method to map retweet chains over the real network and, then, uncover diffusion patterns.

Since we want to analyse information flow on Twitter, i.e. how tweets diffuse on the Twitter Network, we use the transposed graph as described in Section 3.4. As mentioned before, an information chain on Twitter can be represented as a set of related or equal tweets and retweets from different authors. As we saw, a retweet chain is time-dependent because content diffuses on a network of contacts where users can take

[3] http://webgraph.dsi.unimi.it

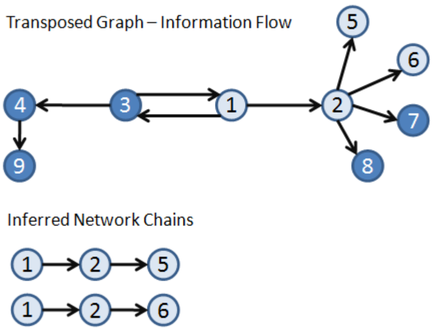

Fig. 5. The result of the first algorithm to infer network chains.

some time to read and diffuse it. Therefore, in what follows, we take into consideration both posting time and how users are connected.

The algorithm to infer network chains from a set of tweets is described as follows:

1. mark all nodes/authors as unexplored;
2. start a Breath First Search (BFS) in the node with the lowest timestamp, i.e. in the root node;
3. proceed with the BFS as usual, ensuring that only nodes that represent the author of some tweet or retweet in the initial set are visited.

Fig. 5 shows the result of the algorithm for the nodes 1, 2, 5 and 6 that participated in some topic with tweets/retweets. Note that node 1 is the root node and the graph is transposed because we are interested in analysing information flow through graph nodes as discussed in Section 3.4. As result of the first algorithm, two sub chains are obtained.

However, a user on Twitter can retweet a tweet from another user without being directly connected to him. For instance, in Fig. 5, if nodes 4 and 9 were also participating in the retweet chain, in the end of the algorithm they would remain unexplored because the algorithm only expands neighbours of a node that also participate in the retweet chain, which does not happen with node 3. So, as a result of the first algorithm, when we have jumps on the network, i.e. users that reference other users in a retweet without being connected to him, there are users that remain unexplored and are not included in any sub chain. Note that inference of retweet chains is very dependent on user network and, thus, it is important to guarantee network quality. However, as mentioned before, the Twitter network is very large and it may be impossible, or take too long, to fully extract. Therefore, it is important to consider that chains are subject to inference errors and, for some users, it is possible that neighbours were not totally explored leading to chain jumps. In order to overcome this limitation, we devise a second algorithm:

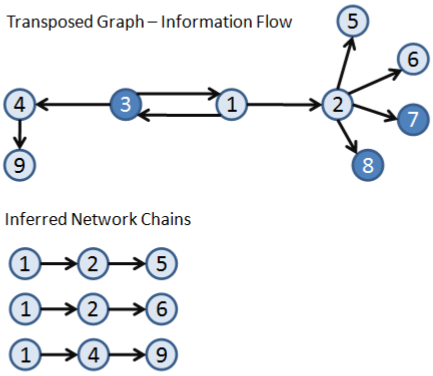

Fig. 6. The result of the second algorithm to infer network chains when there are chain jumps.

1. mark all nodes/authors as unexplored;
2. start a Breath First Search (BFS) in the node with lowest timestamp, i.e. in the root node;
3. proceed with the BFS as usual, ensuring that only nodes that are author of some tweet or retweet in the initial set are visited;
4. if there are unexplored nodes corresponding to authors of tweets or retweets in the initial set, restart the BFS with the unexplored node with lowest timestamp and goto step 3.

With this second algorithm we are able to detect more sub chains as observed in Fig. 6, where nodes 1, 2, 4, 5, 6 and 9 participate in some topic with tweets/retweets and three sub chains are detected. Unlike the first algorithm, nodes 4 and 9 are explored. In this example there is no information about tweet origin in the case of node 4, thus we assume that the source is node 1. Note that in some cases we are still able to identify the source if the author identifies the original author in the retweet.

Inferred network chains provide a set of features that are used to characterize information diffusion processes and user activity. We consider the following features: chain length, i.e. the largest number of sequential nodes in a chain or sub chain; chain width, i.e. the number of sub chains or leaves; chain shape, which can be either a star or tree; chain duration, the time interval between the first tweet that started the chain and its last tweet; user reaction time, i.e. the time interval between the first tweet until a user joins the chain and reacts by retweeting it; number of times a user partic-

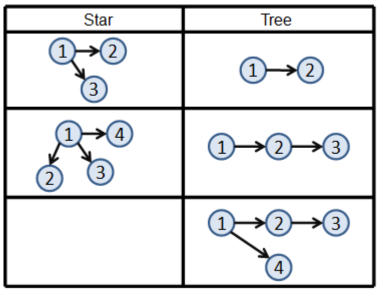

Fig. 7. Stars and trees shapes

ipated; number of times a user participated in chains with star shape; number of times a user participated in chains with tree shape; number of times a user is the first node, or root, in a chain; number of times a user is the first node in a chain with star shape; number of times a user is the first node in a chain with tree shape. Note that there is no clear definition for star and tree shapes and, thus, in this work we define them as follows. A star must have width greater than or equal to two and length equal to two, i.e. a root node with several children, but no grandchildren. A tree can have width one and length one or width greater than or equal to two with length greater than two. Fig. 7 depicts both shapes. Note also that, to measure the number of times a user is root in network chains, we list all chains and sub chains and we count the number of times a given user appears in the first position, i.e. the number of times a given user has the lowest timestamp. To measure the number of user participations in network chains, we count the number of times a given user appears in the list of chains.

3.6. Features

Let us now detail all features considered in our work to perform user classification. We used all features detailed in previous sections, taking into account network information, tweet and retweet diffusion, user activity and context. Thus, we considered as features:

- Network features: *number of followers*, the audience of a user, i.e. how many users subscribe his/her content feed; *number of friends or followees*, the number of users a user follows;
- Activity features: *number of tweets*, the number of tweets a user did; *number of retweets*, the number of times a user diffused information posted by another user; *number of replies*, the number of times a user explicitly interacted with others;
- Contextual features: *number of topics*, the number of different topics in which a user participated;
- Diffusion features, which combine information from both retweet and network chains: *number of times a user was root in retweet chains*, counts how many times a user started retweet chains; *number of participations in retweet chains*, counts how many times a user participated in retweet chains; *number of times a user was root in a network chain with tree shape*; *number of participations in a network chain with tree shape*; *number of times a user was root in a network chain with star shape*; *number of participations in a network chain with tree shape*.

Activity, context and diffusion features were extracted directly from the dataset. The number of topics has a maximum of 33 in our dataset, which is the number of topics used to extract content from Twitter, listed in Table 2. Network features depict node degrees and describe how well users are connected on the network. Activity features describe user behaviour and, as defended by Cha *et al.* [2], news and media tend to do more tweets and retweets and no replies, but users as activists and evangelists tend to do more replies and retweets to maintain their audience. Also, context features can help in distinguishing these type of users, e.g. a user which is an expert in some type of content is expected to be more active in few topics than a news and media user which main objective is to diffuse content from different topics.

Diffusion features are important to classify users since they can show how effective are users to diffuse content, their tendency to join tweet chains and the shape produced by their diffusions. The different shapes are relevant because different diffusion shapes are expected for different type of users. For instance, a news and media user which has a large audience is expected to produce a star. However in the case of a user that does interesting tweets, it is expected that trees get produced because the audience is smaller, but tweets will still reach more people through their neighbourhood.

As mentioned previously, when the network and diffusion features are used, it is important to take into consideration network quality. Also, network features can be incorrectly related with diffusion features because content used to infer retweet chains was extracted before the user network.

3.7. Clustering

Taking into account the features described in the previous section, we applied an Expectation-Maximization (EM) based clustering algorithm. We used the implementation available in Weka[4]. Instead of assigning users to clusters to maximize the differences in means, as in k-means clustering, the EM clustering algorithm computes probabilities of cluster memberships based on a number of probability distributions. Then, the EM clustering algorithm tires to iteratively maximize the overall probability or likelihood of the data given a clustering hypothesis. In the end, the EM method assigns a probability distribution to each user which indicates the probability of it belonging to each of the clusters.

The standard EM clustering method assumes a particular user-defined number of clusters, which we do not know and that we would not like to enforce. This is a well known problem and the implementation of the EM clustering method used in this work allows us to estimate the number of clusters through standard cross-validation.

4. Results and discussion

In this section we present results for about 185 thousand users that participated in retweet chains, in our dataset. As described in previous sections, we computed a collection of features and we applied an EM based clustering algorithm, obtaining 6 well defined clusters. We will start by providing some statistics for the collected dataset and, then, we describe in detail our clustering results and their meaning.

4.1. Dataset statistics

Table 3 provides several statistics that characterize our dataset and that include in particular information about the number of posts, tweets, retweets, replies, users and banned users. Note that these numbers are only for our dataset, collected from November 8th, 2010, until February 10, 2011, and do not reflect the information we get when we access user profiles on Twitter, where for instance the number of tweets is higher.

As mentioned before, we also crawled the Twitter network and we obtained the connections for each one of the 180 thousand users classified in next section.

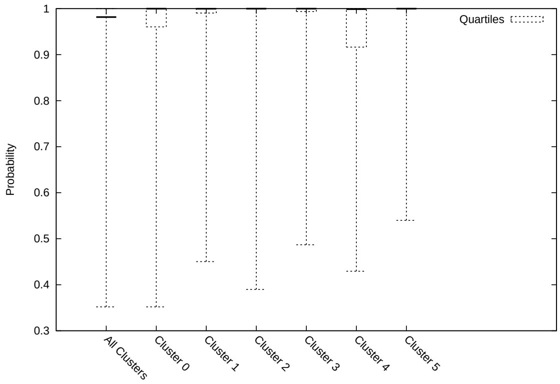

Fig. 8. User probability distribution per cluster.

4.2. User classification

As mentioned before, the EM clustering method returns six clusters, providing for each user the probability of belonging to each cluster. In Fig. 8, we have the user probability distribution for all clusters and per cluster. Although there are users for which assignment probabilities are rather low, more than 75% of users have an assignment probability above 0.9, denoting a well defined cluster assignment. These results show that the selected features allow us to obtain clusterings with high quality, that we would like to validate against some ground truth. It is however hard to evaluate this approach because there is no previous curated data on Twitter user activity. Moreover, such data is hard to obtain, mainly because it is hard to define user activity patterns a priori. The set of features proposed in this paper is a first attempt, and in some sense, orthogonal to previous approaches proposed for Twitter data analysis, and thus can lead to relevant improvements when integrated with other, complementary, features.

Taking into account only users assigned to at least one cluster with a probability above 0.75, we are able to assign 174618 users, about 97% of identified users in retweet chains, distributed as described in Table 4.

We were also able to identify how each cluster is characterized with respect to each feature, depicted in Fig. 9. As discussed before, we do not take into account either profile or content features. We just focus on user activity dynamics and our main goal is to understand how users differ, from a behavioural point of view. Let us then analyse each cluster.

- Users in cluster 0 have a large number of followers and friends, which denotes other users interest in their tweets. Nevertheless, their content is

[4] http://www.cs.waikato.ac.nz/ml/weka/

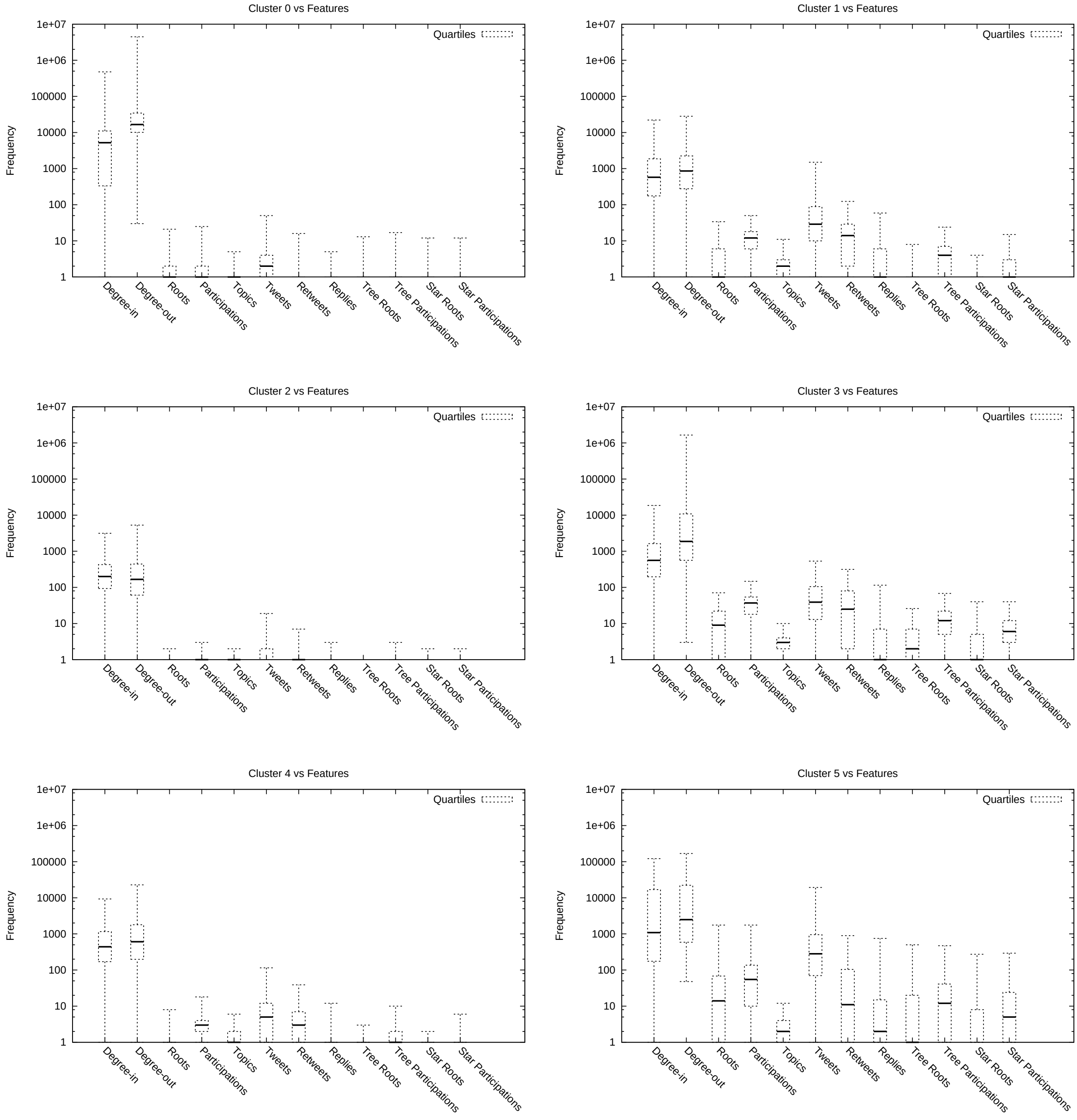

Fig. 9. Features statistical details for each cluster. Note that the y-axis is in log scale.

Table 3

Dataset details before and after filtering for non-Latin characters and banned users

Measure	Result
Before Filtering	
Number of posts	5560492
Number of users	1489888
After Filtering	
Number of tweets	3875648
Number of retweets	548899
Number of replies	341053
Number of users	1350616
Number of banned users	139272
Number of retweet chains	92792
Number of network chains	92479
Number of users in chains	175083
Number of root participations on star network chains	10013
Number of participations on star network chains	62852
Number of root participations on tree network chains	19730
Number of participations on tree network chains	152398

not diffused, since the number of participations in chains as root is very low. We found in this cluster feeds of well known people, such as the Twitter feed *@joshu*— the creator of *Delicious*. We also found contest feeds such as *@MommyPR* and *@heatworld*, about contests and celebrities, respectively. Although different in nature, all these feeds share some characteristics: they attract interest from many people, but their feeds do not generate discussion.

- In cluster 1 we have a set of active users, with a large number of tweets and retweets, as well as a large number of participations in chains. However, given the small number of participations as root, we may conclude that they do not create new chains, i.e. new discussions or influent content. We found in this cluster feeds such as *@LudusTours* about sports events and travels, *@stzlyee* about politics and economics, or *cassandravo* about commercial contests on Twitter. All these feeds have a reasonable number of replies, as users tend to acknowledge, contest earnings or sports travel pictures, for instance.
- Cluster 2 is the largest, with 78.9% of users. It contains users that have very low activity. They post a low number of tweets and, although, they are connected to other users, they do not diffuse content. All examples we found in this cluster are random people or organizations with low activity.
- Cluster 3 contains active users that generate new content in different topics. This conclusion can be drawn by observing the high number of topics, the number of root participations and the number of all participations in chains. Moreover, these users have a large number of followers. Although this cluster includes some news and media, and influential bloggers, it is interesting that the number of replies is high, denoting tendency for chatting and topic discussion. Examples include *@nytimes*, the Twitter feed of *The New York Times*, and *@TheNewDeal*, the feed of the political blog *Big Corporation*.

Table 4

Number of users per cluster

Cluster	Size	%
0	4036	2.3
1	3479	2.0
2	137841	78.9
3	866	0.5
4	28118	16.1
5	278	0.2

- Users in cluster 4 seem to be active and, most of them, retweeters, because the number of retweets is very close to the number of tweets. Moreover, these users tend to not create chains, but nonetheless participate in them. Examples include *@exiphanic* and *@VicksG*, two active Twitter users, and *@chillingo*, the *Chillingo* company. This cluster contains normal users, as in cluster 2, but these users are much more active.
- Finally, cluster 5 contains users who create and participate in a great number of chains from different topics. They have both a large number of followers and a large number of friends. These results show that they may be evangelists, which are users who make, not only influential tweets, but also communicate and work to maintain their audience, as discussed in Cha *et al.* [2]. Examples include *@abc7*, a news feed from KABC-TV, *@declanm* and *@lyn_d75*, both news agencies correspondents.

The cluster sizes also follow the traditional pattern associated with scale-free networks [4], as is the case of Twitter. We have a large cluster, cluster 2, with most users, and a smaller cluster, cluster 4, with a significant number of users. These two clusters together contain almost all users and allow us to distinguish very active users from almost inactive users. The other four clusters have much smaller sizes and, although they contain active users, they differ on activity patterns as observed.

It is important to observe that this classification may be affected by the selected topics (see Table 2). Nevertheless, given the different nature of the selected topics, we believe that we obtained a near real view of Twitter user dynamics. We also note that by selecting different topics, or just a set of tweets related to some subject, our approach will allow user activity classification in that context, which is interesting in itself.

As a final remark, note also that our results allow us to state that the majority of users have human behaviour and are not bots. This can be observed through their reciprocity in the number of followers and friends, an observation done also in Chu *et al.* [3].

5. Conclusions and Future Work

In this paper we propose a set of features that allow us to characterize and distinguish user activity patterns on Twitter. Retweet and network chains are the result the importance attributed by the surrounding neighbours to the content the user posted. Through the analysis of diffusion patterns we are able to infer different kind user behaviour. Note that, our approach does not use information from the user profile. Thus, it is based on what a user does on a network and his/her ability to interact with his/her audience. Our approach was tested in a dataset of about 3870000 posts from 135000 users collected for this effect. We are able to classify 75% of the users in our appropriate cluster with a 0.9 assignment probability. The clusters found follow the traditional pattern associated with scale-free networks. Clusters are fully characterized by a set of statistical features regarding network structure, user activity and dynamic patterns.

Moreover, to analyse the impact of content posted by a user we propose three algorithms, (1) we define an algorithm to extract retweet chains based only on the timestamps and authors of equal tweets and retweets; (2) we propose an algorithm to infer network chains that show how content diffuses on network of neighbours; (3) since users on Twitter can retweet other users without a network connection between them, we propose another algorithm to infer network chains where network jumps are included.

In future we want to combine this approach with other features to detect spammers. Spammers on Twitter tend to do many posts with similar text and URLs, and since normal users are able to detect a spammer and ignore her posts, it is expectable that few retweet and network chains are generated by this type of users.

Moreover, we want to include user profile and content features. Profile features can help to contextualize user behaviour, e.g. it may be possible to detect different behaviour depending on the geographic localization of users. Content features can help to understand how different behaviour can be generated based on what and how users write.

Finally, since there are users that have more ability to diffuse content on specific topics, we want to understand their behaviour.

Acknowledgements

We thank the support FolkPeers Project (PTDC/EIA/73945/2006) from Fundação para a Ciência e Tecnologia (http://alfa.fct.mctes.pt/).

References

[1] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. *Proceedings of the 43rd Hawaii International Conference on Social Systems HICSS*, 2010.

[2] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring user influence in twitter: The million follower fallacy. In *In Proceedings of the 4th International AAAI Conference on Weblogs and Social Media ICWSM*, 2010.

[3] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia. Who is tweeting on twitter: Human, bot, or cyborg? *In Proceedings of the 26th Annual Computer Security Applications Conference*, pages 21–30, 2010.

[4] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World.* Cambridge University Press, 2010.

[5] Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. In *KDD*, pages 1019–1028, 2010.

[6] A. Java, T. Finin, X. Song, and B. Tseng. Why we twitter: Understanding microblogging usage and communities. In *Joint 9th WEBKDD and 1st SNA-KDD Workshop*, 2007.

[7] J. Krauss, S. Nann, D. Simon, K. Fischbach, and P. Gloor. Predicting movie success and academy awards through sentiment and social network analysis. In *16th European Conference on Information Systems ECIS*, 2008.

[8] J. Leskovec, M. Mcglohon, C. Faloutsos, N. Glance, and M. Hurst. Information propagation and network evolution on the web. Unpublished, 2009.

[9] D. R. Millen and J. F. Patterson. Stimulating social engagement in a community network. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work*, pages 306–313, 2002.

[10] N. Naveed, T. Gottron, J. Kunegis, and A. C. Alhadi. Bad news travel fast: A content-based analysis of interestingness on twitter. In *Proceedings of the ACM WebSci'11*, pages 1–7, 2011.

[11] M. Pennacchiotti and Ana-Maria Popescu. A machine learning approach to twitter user classification. In *Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[12] S. Petrovic, M. Osborne, and V. Lavrenko. Rt to win! predicting message propagation in twitter. In *International Conference on Weblogs and Social Media ICWSM*, 2011.

[13] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences).* Cambridge university press, 1994.

[14] J. Yang and S. Counts. Predicting the speed, scale, and range of information diffusion in twitter. In *International AAAI Conference on Weblogs and Social Media*, 2010.