

# Understanding Factors that Affect Response Rates in Twitter

Giovanni Comarela  
Federal University of Minas Gerais, Brazil  
giovannicomarela@dcc.ufmg.br

Virgilio Almeida  
Federal University of Minas Gerais, Brazil  
virgilio@dcc.ufmg.br

Mark Crovella  
Boston University, USA  
crovella@cs.bu.edu

Fabricio Benevenuto  
Federal University of Ouro Preto, Brazil  
fabricio@dcc.ufmg.br

## ABSTRACT

In information networks where users send messages to one another, the issue of information overload naturally arises: which are the most important messages? In this paper we study the problem of understanding the importance of messages in Twitter. We approach this problem in two stages. First, we perform an extensive characterization of a very large Twitter dataset which includes all users, social relations, and messages posted from the beginning of the service up to August 2009. We show evidence that information overload is present: users sometimes have to search through hundreds of messages to find those that are interesting to reply or retweet. We then identify factors that influence user response or retweet probability: previous responses to the same tweeter, the tweeter's sending rate, the age and some basic text elements of the tweet. In our second stage, we show that some of these factors can be used to improve the presentation order of tweets to the user. First, by inspecting user activity over time, we construct a simple on-off model of user behavior that allows us to infer when a user is actively using Twitter. Then, we explore two methods from machine learning for ranking tweets: a Naive Bayes predictor and a Support Vector Machine classifier. We show that it is possible to reorder tweets to increase the fraction of replied or retweeted messages appearing in the first  $p$  positions of the list by as much as 50-60%.

## Categories and Subject Descriptors

J.4 [Computer Applications]: Social and behavioral sciences Miscellaneous; H.3.5 [Online Information Services]: Web-based services

## General Terms

Human Factors, Measurement

## Keywords

Social Networks, Twitter, timeline

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

HT'12, June 25–28, 2012, Milwaukee, Wisconsin, USA.

Copyright 2012 ACM 978-1-4503-1335-3/12/06 ...\$10.00.

## 1. INTRODUCTION

Twitter has become a vast network, hosting an immense flow of information passing among its users. According to [1], Twitter users now send more than 140 million messages (*tweets*) per day, meaning that Twitter carries a billion messages every 8 days. As a consequence, a flood of information swamps Twitter users. For instance, according to [7], active users can easily receive more than 1000 tweets a day. It is difficult or impossible for a user to keep up with this amount of tweets. However, many tweets are irrelevant, superfluous, or too difficult to understand without context. For example, the recent study presented in [2] estimates that only 36% of the Twitter's feed are worth reading. Clearly, users can benefit from tools that help them sort the "wheat from the chaff."

To do so, a starting point is to understand how users consume messages and interact with other users in the network. Research on Twitter is newer than on e-mail and Web so we know less about the behavior of users in this information network. A fundamental set of questions then is the following: how do users manage the incoming flood of messages in Twitter? How do users value incoming messages, as evidenced by their generation of message responses (*replies* and *retweets*)? What factors affect whether users reply to or retweet messages? Answers to these questions can help us reason about management strategies to distinguish important or interesting messages.

Our approach to answering these questions begins with an extensive characterization of the behavior of Twitter users. To understand user behavior, we looked at a very large dataset, consisting of all tweets exchanged among Twitter users over more than two years. Our focus is on understanding and improving the timeline of tweets presented to the user. When users log in to Twitter, they typically see a chronological stream of tweets as sent by all of their sources. Users interact with this tweet-stream by replying to tweets, or resending tweets (retweeting). The tweets a user reply to or retweet thus can provide indication of a sense of importance or *interestingness* the user ascribes to the specific tweet. Hence we reconstructed the timeline for a numbers of subsets of users and examined questions related to the reply and retweet behavior of users.

Our first set of findings consists of characterizations of reply and retweet behavior. From the outset it is clear that information overload is a common experience of Twitter users; we show that users sometimes have to search through hundreds of messages to find those that are important. We then identify features that influence user response or retweet probability: previous responses to the same tweeter, the tweeter's sending rate, the age of the tweet, the size of the text message and the presence of *mentions*, *hash-*

*tags* or *URLs*. We show that these features are correlated with the likelihood that a user will reply to or retweet a tweet.

These findings not only unveil unique aspects of user behavior and their interactions in Twitter, but they also motivate us to propose an approach to improve a user’s interaction with Twitter by reordering the presentation of tweets in a user’s timeline. This prompts our second set of findings: by inspecting user activity over time, we construct a simple on-off model of user behavior that allows us to infer when a user is actively using Twitter. Then we investigate the feasibility of applying machine learning methods to rank tweets in a user’s timeline in a manner that brings the most important tweets to the top of the timeline. We take two different approaches: a Naive Bayes predictor that combines the empirically observed conditional response probabilities, and a Support Vector Machine classifier. These methods do not make use of the contents of tweets, and so are lightweight enough for implementation in Twitter clients. Using trace-driven simulation we show that employing only three content-independent features, it is possible to reorder tweets to increase the fraction of replied or retweeted messages appearing in the first  $p$  positions of the list by as much as 50-60%. These results indicate that our methodology can originate an interesting alternative interface to present tweets instead of the usual reverse chronological order.

The remaining of this work is organized as follows. Section 2 describes related work. In Section 3 we present our dataset. Section 4 discusses characteristics associated with response rates in Twitter. In Section 5 we show machine learning based approaches to reorder the common Twitter *timeline*, and Section 6 provides an evaluation of these algorithms. Finally, Section 7 summarizes the paper and presents the main findings of this work.

## 2. RELATED WORK

User behavior characterization is fundamental to the understanding and engineering of efficient information networks. Next, we survey studies that are focused on analyzing user characteristics and interactions among users.

There has been a number of studies that provide an overview of Twitter and its users. In [19], the authors present a detailed characterization of Twitter. They gathered three datasets (covering nearly 100,000 users) and identified distinct classes of Twitter users and their behaviors, geographic growth patterns and current size of the network. The work in [17] studies linked structures of social networks and argue the structures do not reveal actual interactions among people. Scarcity of attention and the daily rhythms of life and work drive interaction patterns in social and information networks. This study of social interactions within Twitter reveals that the driver of usage is different from the declared network of friends (i.e., followings). The authors of [18] present a characterization of the Twitter phenomena and study topological and geographical properties of the information network. They show people use messages to talk about their daily activities and to seek or share information. They also analyze user interactions at a community level.

Several authors present data-driven analysis and measured patterns of communications as well as information spreading across social network links. Kwak *et al.* [20] provides an in-depth characterization of different aspects of Twitter. In particular, they found a non-power-law for follower distribution, a short effective diameter, and low reciprocity. These characteristics contrast with known characteristics of human social networks. A prominent model of human behavior related to the amount of information one can process was presented in [5]. Their approach is based on a queuing model with a priority discipline and assumes that every individ-

ual prioritizes different activities and executes the corresponding task with the highest priority. With this assumption, this model shows that the waiting time of tasks follows a power-law distribution. In [11] the authors study the reasons that make a conversation to be interesting or what prompts a user to participate in the discussion on a social network. They conjecture that people participate in conversations when they find the conversation theme interesting or when they see comments by people they know or when they observe an engaging dialogue between two or more people. The reference then introduces the concept of *interestingness*. They also propose a mathematical framework to measure the *interestingness* of textual conversations about a certain video on YouTube. They collect commentaries on YouTube videos and show that their method to measure *interestingness* of a conversation provides a better assessment than traditional methods, based on the number of comments, number of new participants, etc.

Bernstein *et al.* [7] presents an approach to group tweets, by identifying a topic a tweet belongs to. Finally, there has been recent efforts on identifying influential users in Twitter [4, 9, 25]. One of the possible use of these efforts is to sort upcoming tweets according to sender’s influence score. However, current influence metrics are susceptible to be fooled by spammers [6], bots [8] and social capitalists [14] and do not capture the temporal dynamics of Twitter. One important observation from [9] is that highly influential users are not necessarily the most followed users, meaning that aspects of the Twitter topology are not sufficient to capture one’s influence. Regarding the attention that users pay in their Twitter’s timeline, a recent effort [13] used eye-tracking techniques to measure which tweets has more user’s attention. Among their findings, they show evidence that reply patterns reflect attention interest. They also show that only tweets above a relatively high threshold in terms of attention and interest are considered for retweets.

Compared to this body of work, our work take different directions as our interest is mainly on the study of user interactions in Twitter. We also are interested at finding better mechanisms to organize incoming streams of tweets. To this end, we measure and model user behavior aiming at capturing how interesting a tweet can be for a user and how to use these informations in a simple process in order to reorganize timelines. To keep the procedure simple, we decided do not look at the tweet’s content because, in general, topic identification is CPU intensive (not desirable for handheld devices). Moreover, studies of interactions on Twitter are generally more focused on retweets, for example [24], and here we also present an extensive study for replies, which are an important type of interaction, but less explored in the literature.

## 3. TWITTER DATASET

Our dataset contains extensive data from a previous measurement study that included a complete snapshot of the Twitter network and the complete history of tweets posted by all users from 2006 to July 2009 [9]. Our dataset contains 54,981,152 user accounts connected to each other by 1,963,263,821 social links. Our dataset also contains all tweets ever posted by the collected users, which consists of 1,755,925,520 tweets. For detailed characteristics of this dataset we refer the user to [9, 23].

This dataset is very suitable for the purpose on this work as it contains all tweets exchanged among all users over a long period of time as well as the social links among users. From the tweets in our dataset we can identify two special types of messages that a user can post: *replies* and *retweets*. A *reply* occurs when a user replies other tweet posted by other user. A *retweet* refers to the practice of copying a tweet of someone and post it with a personal comment (optional). In our dataset each *reply* has the IDs of the tweet and

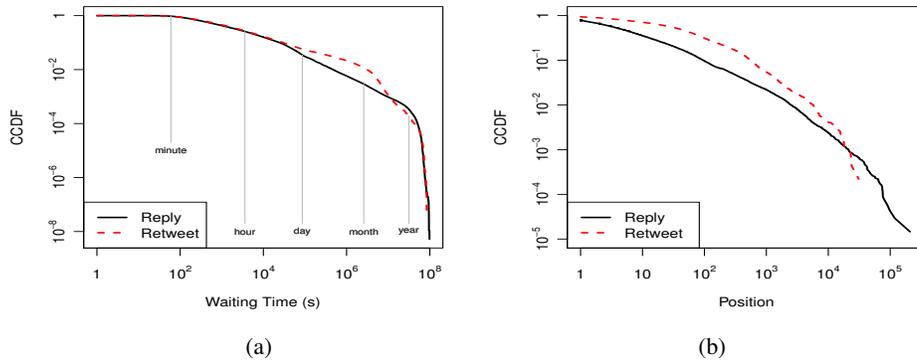


Figure 1: (a) Waiting time and (b) timeline position distributions for replies and retweets.

user that were replied. In the case of *retweets*, on the other hand, we do not have such information, since this type of message was done through a user’s convention. In this convention the users used the pattern RT @user\_name with the content of the original message plus a comment, where user\_name is a name that is uniquely assigned to each Twitter user. We used this information to identify the origin (user and tweet) of each *retweet* in the dataset. Moreover, a tweet can have two kinds of elements embedded: *mentions* and *hashtags*. A *mention* is a citation of another user and a *hashtag* is a way to mark keywords in a tweet. These elements are respectively used through the patterns @user\_name and #keyword.

## 4. CHARACTERIZING INTERACTIONS

Our first set of results consists of a characterization study of the behavior of Twitter users. This section presents those results as a foundation for our proposed timeline ordering algorithms, which are presented later.

### 4.1 Notation and sampling setup

We start by introducing notations and definitions. We use  $U$  to denote the set of all Twitter users in the dataset. For each  $i \in U$ , the total of messages sent by  $i$  is  $T_i$ , and the  $j$ -th message sent by  $i$  is  $m_{j,i}$ . We denote by  $M_i = [m_{1,i}, \dots, m_{T_i,i}]$  the list of all messages posted by  $i$  in chronological order, and  $M_V = \bigcup_{j \in V} M_j$  (also in chronological order) the list of all messages sent by the set of users  $V$ . The set of all users followed by user  $i$  is denoted  $Out_i$ , and the set of all users followed by any user in set  $V \subset U$  is  $Out_V = \bigcup_{j \in V} Out_j$ . For any user  $i$ , the list of messages sent by users in  $Out_i$ , presented in reverse chronological order, is user  $i$ ’s timeline, denoted by  $TL_i$ .

Many of our characterizations are based on user’s timelines. Unfortunately it was not feasible to construct timelines for all users in the dataset (the dataset size has almost 1 TB). Instead we took random samples from  $U$ , and built timelines for those users.

Sampling users gave us another advantage. Because the level of user activity varies dramatically from user to user, it is important to consider users with different activity levels separately. Thus, we took four samples, each one comprising 2000 users. These samples, denoted by  $S_1, S_2, S_3$  and  $S_4$  were sampled at random from those users that posted more than 2, 10, 100, and 1000 messages respectively. As it will be seen below, users belonging to different sets often show quite different characteristics. When a result is not associated with one of those specific samples, it refers to the entire dataset (all users in  $U$ ).

After this step we extracted from the dataset the users that are

followed by users in each of the four samples ( $Out_{S_k}$ ) as well as their tweets ( $M_{Out_{S_k}}$ ). In this way we were able to build all timelines,  $TL_i, \forall i \in S_k$ . Finally, we also extracted the message sets  $M_{S_k}$  from the data for  $k = 1, \dots, 4$ .

### 4.2 Waiting Times

We start our characterization study by looking at evidence that Twitter users experience information overload. We do so by examining the duration between when a tweet arrives in the user’s timeline and when it is replied to or retweeted.

Figure 1(a) shows the Complementary Cumulative Distribution Function (CCDF) for the time that a tweet waits to be replied or retweeted, across all replies and retweets in the dataset. We can see that almost all messages wait more than 100 seconds until they are replied or retweeted and that approximately 90% can wait up to 1000 seconds. Figure 1(b) shows the distribution of the position in the user’s timeline of replies and retweets for sample  $S_1$ . This figure shows that sometimes users must search far back in their timeline to find a tweet that they wish to retweet or reply to. For example, 10% of the retweets are made to tweets that are more than 800 positions back in the user’s timeline.

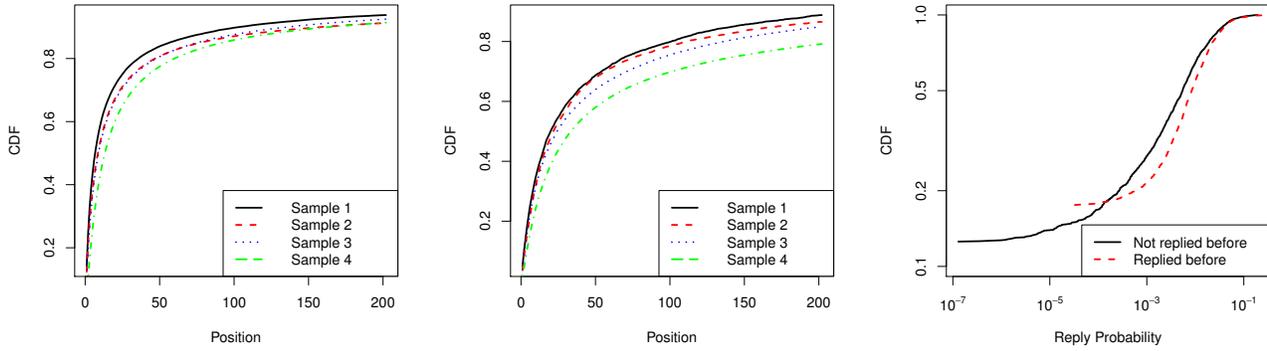
This motivates efforts to change the presentation order of the user’s tweets in such way that messages will wait less time before being replied or retweeted.

Figure 1(a) shows other important points. First, it shows that after one day, the behavior of waiting times for replies and retweets becomes different. This shows that message replies tend to occur more quickly than retweets — suggesting that users are more likely to share old information than to reply to old messages. This is confirmed by Figure 1(b) which shows that users do not search as far back in their timelines for tweets to reply to as they do for retweets. Second, this figure suggests a general change in user behavior around the timescale of one day. Finally, we can note that both curves from Figure 1(a) drop off sharply near  $10^8$  seconds. This corresponds to the age of the Twitter system at the time that the dataset was collected.

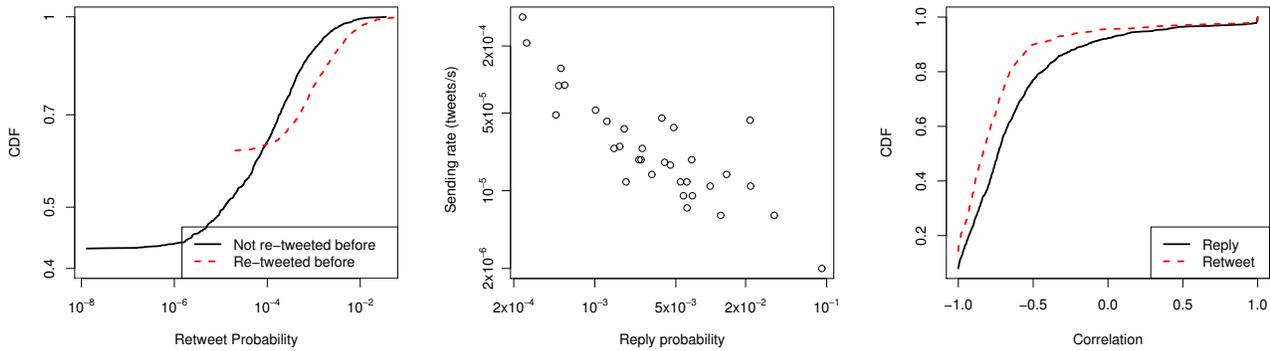
### 4.3 Tweet Age

Having motivated a search for better ordering of tweets, we turn to features that may help in assessing their importance. The first feature we consider is the age of a tweet. In particular, we ask: *are newer messages more likely to be replied or retweeted?*

To answer this question we return to the question of where in the user’s timeline each replied (retweeted) message was found. In other words, for  $j = 1, \dots, T_i$ , if  $m_{j,i}$  is a reply (retweet) we looked for the message it replies to (retweets) in  $TL_i$ . We do this



(a) CDF of the Position in the timeline of replied tweets (b) CDF of the Position in the timeline of retweeted tweets (c) CDF of the conditional reply probability for users in sample  $S_3$



(d) CDF of the conditional retweet probability for users in sample  $S_3$  (e) Reply probability  $\times$  Sending rate of the Senders (specific user) (f) CDF of Linear Correlation between  $\log(\text{Sending rate})$  and  $\log(\text{reply/retweet probability})$

**Figure 2: Characterizations of attributes related with the likelihood of interaction with tweets in the user’s timeline.**

with respect to the state of  $TL_i$  at the moment that  $m_{j,i}$  was posted, and note the position of the message (with the newest message assigned position 0, etc.). We compute this quantity for each user  $i \in S_k, k = 1, \dots, 4$ .

Figures 2(a) and 2(b) present the Cumulative Distribution Function (CDF) of the position in the timeline of replied and retweeted messages. The distributions show that newer messages are much more likely to generate responses than older messages. For example, it can be seen that 14% of replies are in position 0 in  $S_1$ . This fraction decreases to 12%, 10% and 5% for  $S_2, S_3$  and  $S_4$ , respectively. Moreover, 84% (81%, 80%, 77%) of all replies happen in the top 50 for  $S_1 (S_2, S_3, S_4)$ .

The same phenomenon is found for retweets. The figure shows that 68% (68%, 64%, 58%) of all retweets are in the top 50 for  $S_1 (S_2, S_3, S_4)$ . Moreover, both figures show that more active users have a higher probability of replying or retweeting messages at higher (older) positions, indicating that users who send more tweets also spend more time reading and interacting with their received messages.

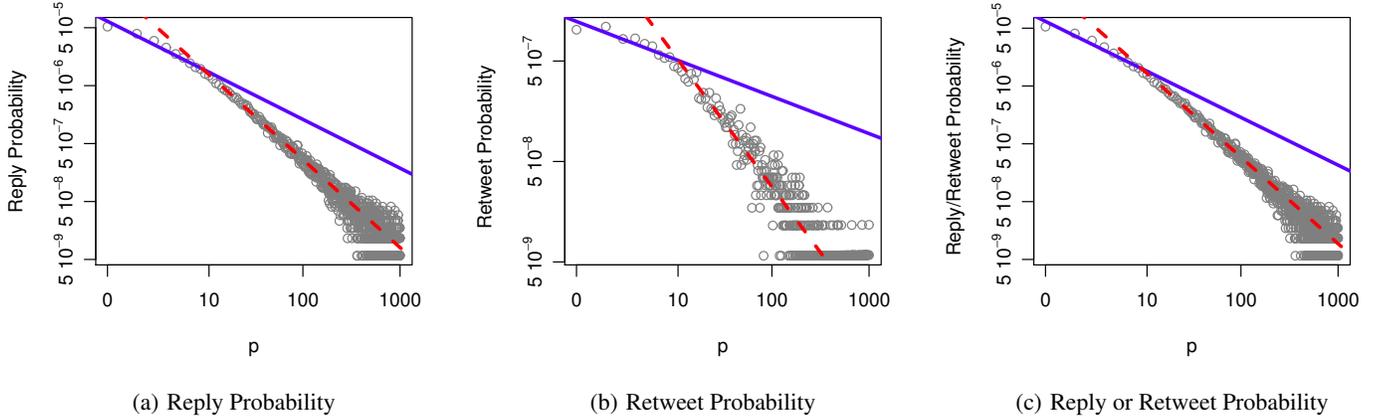
To complement this result we decided to compute the probability that a user  $i$  will reply (retweet) a tweet  $m$  given it is in position  $p$  of his/her timeline. If  $m$  is in the position  $p$ , then  $i$  had the chance to interact with  $m$  when it was in the positions  $0, \dots, p-1$  and has now the chance to interact with it in the position  $p$ . In this way, we

computed the fraction of tweets replied or retweeted in a specific position  $p$  from all tweets that could be replied (retweeted) in this position. Figure 3 presents the probability for replies, retweets and for both of them as a function of  $p$  for sample  $S_1$ . In these three cases we can see two different shapes for the curves, one in the beginning (for  $p \leq 10$ ) and other in the tail ( $p > 10$ ) and that in both cases a linear function can be a good approximation (in logarithmic scale). The parameters of the fitted curves were obtained through a linear regression. This same behavior of two shapes was observed in all other samples.

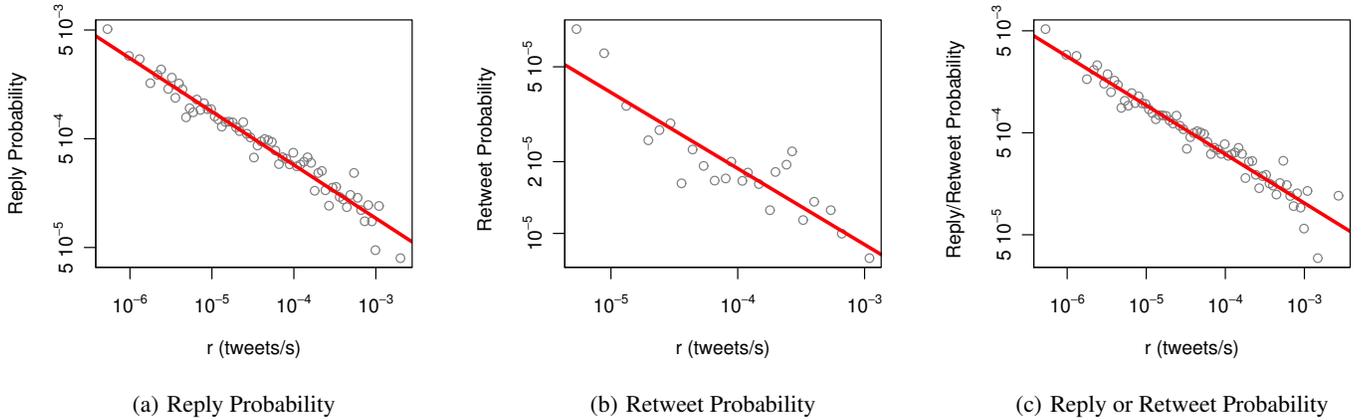
#### 4.4 Prior Interaction

The next feature we consider is whether the tweet sender has previously sent a tweet that was retweeted or replied to. Thus we ask: *are previously replied (retweeted) users more likely to have their tweets replied to (retweeted) again?* To answer this question we proceed as follows: For each user  $i \in S_k (k = 1, \dots, 4)$  and for each message  $m \in TL_i$  we compute the conditional probability that  $m$  will be replied (retweeted) given that the sender of  $m$  was replied (retweeted) by  $i$  before. We also compute the same probability given the complementary event.

For example, to compute the probability that  $i$  replies to a user given that  $i$  has replied to that user before, we count two events for each message that  $i$  has received:  $A$ , the number of times that  $i$



**Figure 3: Probability of (a) Reply, (b) Retweet and (c) Reply or Retweet a tweet given that it is in the position  $p$  of the user's timeline.**



**Figure 4: Probability of (a) Reply, (b) Retweet and (c) Reply or Retweet a tweet given that it was sent by a user with sending rate  $r$ .**

replies to a message that comes from a previously-replied-to user, and  $B$ , the number of times that a received message comes from a previously-replied-to user. We then estimate the corresponding probability as  $\frac{A}{B}$ .

Figures 2(c) and 2(d) show the CDF of the conditional probabilities computed for all users of the sample<sup>1</sup>  $S_3$ . The figures show that for replies (retweets), the probability distribution of a reply (retweet) rate increases (shifts to the right) when the sender has been replied-to (retweeted) before. The figure shows that for retweets, this shift is approximately by a factor of 5. In other words, in general, a user is 5 times more likely to retweet a sender that the user has retweeted before, than if the sender has not been retweeted before.

#### 4.5 Sender Rate

The third feature we examine is the activity level of the sender – how prolific the sender is in sending tweets. In this section we seek to answer the question: *Do more active users have a higher chance to be replied (retweeted)?*

To answer this question, we find for each user  $i \in S_k$  ( $k = 1, \dots, 4$ ) the users that  $i$  follows,  $j \in Out_i$ . For each such  $j$ , we compare their sending rate with the fraction of their sent messages

<sup>1</sup>We do not present the figures for all four groups  $S_3$  due to space restrictions.

that were replied (retweeted) by  $i$ . We define the sending rate of user  $j$  as  $T_j$  divided by the time between the first and last messages posted by user  $j$ . It is important to remark that this variable considers at once the number of tweets of a user and the interval of time that the user is active in the network.

Figure 2(e) shows a scatter plot of these variables for an example user  $i$ . Each point corresponds to a user  $j \in Out_i$ . Users who were never replied to by  $i$  are omitted due to the log scale. The figure shows that the higher the sending rate of the user, the lower the reply probability. Moreover, this figure shows a strong linear correlation in log scale. To verify if this relationship holds across all users, we repeated this procedure for each  $i \in S_k$  and computed the linear correlation between these two variables (sending rate and reply probability) in log scale. Figure 2(f) shows the results for replies and retweets for sample  $S_2$ . It can be seen that for replies (retweets) almost 80% (90%) of all users have a correlation coefficient smaller than -0.5, while only 10% (9%) have a positive correlation. Thus, we find that almost all users are more likely to reply or retweet a sender if his/her sending rate is low.

In order to understand better this matter we also computed the probability that a user  $i$  will reply (retweet) a tweet  $m$  given that it was posted by a user with a sending rate of tweets  $r$ . To this end, for all users  $i \in S_k$  we computed the fraction of replies (retweets) to users with sending rate of  $r$  over the total number of received tweets that came from senders with this same sending rate. As this is a

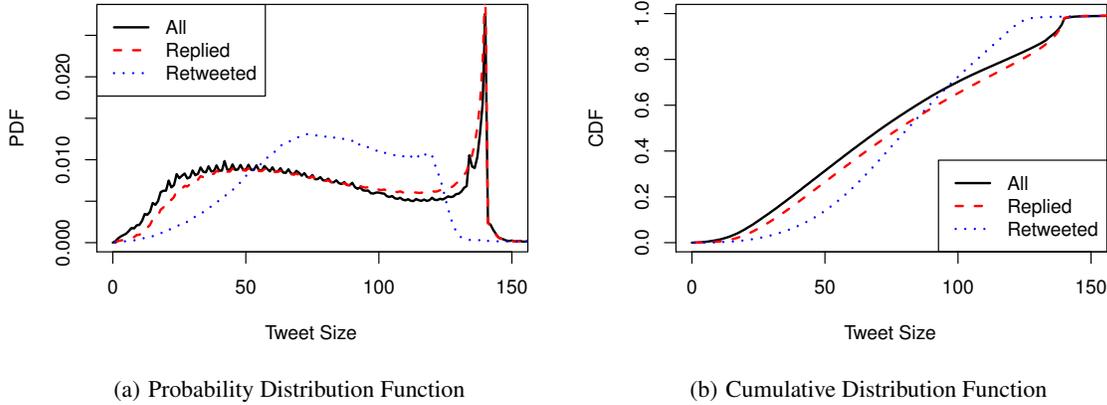


Figure 5: Characterization of the importance of the tweet size in the likelihood of interaction with tweets in the user’s timeline.

continuous variable, we aggregated their values in logarithmic bins and grouped bins to ensure that each bin had at least one hundred of replies (retweets). Figure 4 presents probabilities in function of  $r$  and the fitted curve for sample  $S_1$ . Clearly this figure maintains the conclusions obtained with Figure 2(f).

#### 4.6 Message features

In this section we investigate the influence of intrinsic characteristics of Twitter messages on the interaction among users, represented by replies and retweets. We do not look into the semantic content of tweets. Rather, we analyze the size of tweets, the presence of *hashtags*, *mentions*, and embedded links (i.e., *URLs*). In order to do that we look at such characteristics in all tweets of our data set, i.e. we do not use samples. We considered three separate datasets: *i) Replied* tweets; *ii) Retweeted* tweets; and *iii) All* tweets.

Table 1 displays the fraction of tweets with hashtags, mentions and URLs in these three datasets. On one hand, we see that tweets with hashtags and URLs are more likely to be retweeted. On the other hand, we observe that the same behavior does not hold for tweets with mentions, once only 25% of the retweeted messages contain a specific mention. The characteristics of the Replied dataset are exactly the opposite, the most of the replied tweets (i.e., 55%) contain mentions. Only a small fraction of the replied tweets have hashtags or URLs.

Table 1: Fraction of tweets with hashtags, mentions and URLs.

|         | Replied     | Retweeted   | All  |
|---------|-------------|-------------|------|
| Hashtag | 0.04        | <b>0.16</b> | 0.05 |
| Mention | <b>0.55</b> | 0.25        | 0.36 |
| URL     | 0.10        | <b>0.51</b> | 0.22 |

Does size matter for retweets or replies? Tweets can be up to 140 characters in length. Our findings show that retweeted messages have a positive relationship to sizes considerably smaller than 140 characters. Figures 5(a) and 5(b) show the Probability distribution (i.e. PDF) and the Cumulative distribution (i.e., CDF) curves of tweet sizes for the three datasets considered above. The first observation is that we find in the dataset some tweets with more 140 characters. There were only few cases and we conjecture that the reason is some interface failure in the beginning of the service.

By examining figures 5(a) and 5(b) we notice that the PDF and CDF curves for the *Replied* and *All* datasets are similar, which means the size characteristics of replied tweets do not differ from

the rest of the tweets. However, the *Retweeted* set shows strong differences when compared to the rest of the tweets. Figure 5(a) shows no specific peak near 140 for retweets. Instead it exhibits a plateau ranging from 50 to 120 characters. Shorter tweets were retweeted much more often than longer tweets. One possible explanation is that shorter tweets leave followers with more room add names or personal comments they want to make. In other words, short tweets make possible the participation of different users in the retweeting process.

#### 5. REORDERING THE TIMELINE

The previous section showed that Twitter users can spend a long time searching through tweets in their timelines until they find one interesting enough to reply to or retweet. It also showed some characteristics that indicate which type of tweets are most interesting to users. Motivated by these results, in this section we present two algorithms to reorganize the Twitter timeline. The goal is to present the most interesting tweets for a user first.

Although we presented seven characteristics related to the interaction rate on Twitter, we use only three in our methodology: tweet age, sending rate of the sender and prior interactions. We proceed in this way once our goal is to have a general methodology, good for replies and retweets and as we saw the text features which we studied are not good for both at the same time. Moreover, the authors of [2] showed through a qualitative study that the “bad” use of some textual elements can make tweets “boring” in the user’s opinion.

The two approaches we describe in this section are instances of the general procedure described in Algorithm 1. The main idea is to recognize that users can be in two different states regarding their interaction with Twitter: online (or ON), when they are paying attention in their timelines; and offline (or OFF), otherwise. When a user is in the ON state, the user is viewing all the tweets that have been received during the user’s last OFF session, as well as tweets received while in the ON state. Based on this typical user’s behavior we perform the timeline reorganization upon occurrence of either of two events: *i) every change of state from OFF to ON* (Line 2); and *ii) every tweet arrival* (Line 5). After the reorganization process the revised timeline  $TL'$  is presented to the users instead the old one ( $TL$ ).

The instances of this algorithm are obtained with two different versions of the procedure *Reorganize*. Both are based in machine learning techniques used to compute score for tweets based on their characteristics discussed in the last section. These techniques are

---

**Algorithm 1:** General Timeline Reorganization

---

**Data:** user  $u$  and Tweets of  $u$  received in the last OFF session

- 1  $TL \leftarrow$  Set of all Tweets received in the last OFF session
  - 2  $TL' \leftarrow \text{Reorganize}(TL)$
  - 3 **foreach** Tweet  $m$  received in the current ON session **do**
  - 4      $TL \leftarrow TL \cup \{m\}$
  - 5      $TL' \leftarrow \text{Reorganize}(TL)$
  - 6 **end**
- 

the *Naive Bayes* predictor (NB) and a *Support Vector Machine* classifier (SVM) and will be presented in Sections 5.2 and 5.3 respectively. After this step of computing scores, the tweets are sorted in such way those that are more likely to interact with will be presented first.

One important characteristic of our methodology is that it considers the change of score that tweets can have over the time, once the *Reorganize* procedure can be performed several times during an ON session. In such way, a tweet which is interesting now may not be in the future. In practical terms, it can be very expensive to reorder the timeline every time that a new tweet arrives. In order to avoid this problem the second type of event can be replaced by some *time out* mechanism or the arrival of at least  $l > 0$  tweets in the timeline.

Since our dataset does not allow us to know when each user was ON or OFF we decided to use a simple model to infer this user's behavior. This model is presented in the next section.

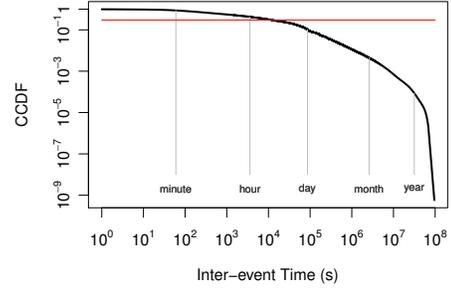
## 5.1 ON-OFF Model

In this section we present a simple model to describe the interaction between Twitter and its users. As described in the last section, each user can be ON or OFF. In our model, an ON session is defined as the interval of time during which the user is actively engaged in posting tweets to Twitter, in such way, that the time between two consecutive messages posted does not exceed a preset threshold  $T_{OFF}$ . In the remaining of the time we say that the user is in an OFF session. We observe that our definition of user interaction session does not include passive activities, such as reading messages.

Since the dataset does not explicitly identify the delimiters of a given session, the number of sessions in the Twitter dataset depends on our choice of the  $T_{OFF}$  parameter. Following the procedure presented in [21] we varied this timeout value  $T_{OFF}$  and counted the total number of sessions. A value extremely small (e.g. 1 minute) could result in a high volume of sessions. As the value of  $T_{OFF}$  increases, the number of sessions is reduced continuously until stabilizing. Following this procedure we considered only users with more than a thousand of tweets. We did in this way because users with few tweets could increase the variance of the total number of sessions, leading to a non precise estimate of  $T_{OFF}$ . The result of this measurement showed that the number of sessions stabilizes when  $T_{OFF}$  is near to  $10^4$  seconds, which represents almost three hours.

Compared to previous efforts, which characterize sessions in traditional Web sites [3, 22], the timeout values obtained are much longer compared to the 10-45 minutes usually observed. The most intuitive reasons for this behavior are the longer time period that users take on Twitter, keeping a track of real time events all the time. This high value of  $T_{OFF}$  is also supported by Figure 6, which presents the Complementary Cumulative Distribution Function (CCDF) of the inter-event (we consider an event any tweet

posted) time of all users in our dataset. In this figure we can see that 70% of all inter-event times are smaller than  $10^4$  seconds.



**Figure 6:** CCDF for inter-event time for all tweets in the dataset. Above the horizontal line we have 70% of the observations.

It is important to remark that this model is an heuristic to infer when users are ON or OFF. We adopted this strategy due to our dataset limitations. In a real situation, if there is better information about the user's state, it should be used.

## 5.2 Naive Bayes approach

This section aims at presenting how we use the *Naive Bayes* predictor [15] to assign scores to tweets in the procedure *Reorganize* of the Algorithm 1. We represent this score through a probability measure for each tweet, which gives the probability that a user will interact with that tweet given a set of its attributes. As previously said we consider three attributes:

- $Age(m)$ , the age of the tweet  $m$  represented by its position in the timeline (the newest one has  $Age(m) = 0$ , the next  $Age(m) = 1$  and so on);
- $SR(m)$ , the average sending rate of tweets of the users that has sent  $m$ ; and
- $I(m)$ , a binary indicator which can be 1 if the user has interacted with the sender of  $m$  before and 0 otherwise.

In this way, for each tweet  $m$ , its score, denoted by  $P(m)$ , is defined as follows:

$$P(m) = P(\text{Interact with } m \mid Age(m) = p, SR(m) = r, I(m) = b).$$

Under the naive assumption we consider the independence among the events  $Age(m) = p$ ,  $SR(m) = r$  and  $I(m) = b$ , which leads us to:

$$\begin{aligned} P(m) &= P(\text{Interact with } m \mid Age(m) = p) \\ &\times P(\text{Interact with } m \mid SR(m) = r) \\ &\times P(\text{Interact with } m \mid I(m) = b). \end{aligned}$$

We present next how we estimated each one of these probabilities in the next three sections.

### 5.2.1 Age score

In Section 4 we presented the probability of replying (retweeting) a message  $m$  given  $Age(m) = p$  in Figure 3. Motivated by this figure we decided to use the following model:

$$P(\text{Interact with } m \mid Age(m) = p) = \begin{cases} \beta_1 p^{\alpha_1}, & p \leq 10 \\ \beta_2 p^{\alpha_2}, & p > 10. \end{cases} \quad (1)$$

It is important to remark that during the fitting process, in all cases the  $R^2$  of the linear regression were greater than 0.81, for retweets, and 0.95 in the other cases.

### 5.2.2 Sending Rate Score

Proceeding as in the previous section and motivated by Figure 4 we assume for the sending rate probability the following model:

$$P(\text{Interact with } m \mid SR(m) = r) = \beta r^\alpha. \quad (2)$$

The  $R^2$  of the fitted curves in all cases are greater than 0.8, for retweets, and 0.94 in the other cases.

### 5.2.3 Interaction Score

Once  $I(m)$  is a binary variable we decided to use the following model for this score:

$$P(\text{Interact with } m \mid I(m) = b) = \begin{cases} \gamma_1, & b = 1 \\ \gamma_2, & b = 0, \end{cases} \quad (3)$$

where  $0 \leq \gamma_1, \gamma_2 \leq 1$ .

In order to estimate these parameters, for all  $u$  in a subset of  $U$  we proceed in the following way: for each arrival in his/her timeline we compute the fraction of tweets  $m$ , that were replied (retweeted) and had  $I(m) = b$  over all received tweets with  $I(m) = b$ .

## 5.3 Support Vector Machine Approach

Support Vector Machine (SVM) is a set of useful methods widely used in data classification to recognize patterns. The most common version of SVM is a binary supervised classifier that maps a vector of attributes in two classes. In this paper we used this SVM version, more specifically the presented in [16], to implement the *Reorganize* procedure of the Algorithm 1. The idea is composed of two steps. First, we consider the same attributes used in the previous section in order to classify the tweets of the timeline in interesting (most likely to interact with) and non interesting. In others words, for each tweet  $m$  in the user's timeline we use SVM to map the vector  $[Age(m), SR(m), I(m)]$  in 1 (interesting) or 0 (non interesting).

Second, we present for the user all tweets classified as 1, in reverse chronological order, and after, all classified as 0, also in reverse chronological order. It is important to remark that as the attributes of a tweet can change over the time, the same can happen with its class. Details about the training phase of this classifier are presented in Section 6.1.

## 6. EXPERIMENTS

In this section we present the results of the proposed algorithms. First we show the evaluation methodology and then we present the performance of the algorithms. After, we show the impact of parameter  $T_{OFF}$ , of our ON/OFF model, in the results and finally, we discuss the difference in the performance of the algorithm when we apply it for users' classes with different activities' patterns.

### 6.1 Evaluation Methodology

In order to evaluate whether the proposed algorithms are effective we performed a trace-driven simulation to figure out what would happen if the tweets were presented according to our methodology instead of the reverse chronological order (in the official Twitter interface). The necessary steps for this simulation are presented in Algorithm 2 and explained below.

For each user  $u$ , we split all his/her tweets in ON and OFF sessions according to Section 5.1 (Line 1). After that, for each tweet which is an interaction of  $u$  (an interaction here can be a reply or

---

### Algorithm 2: Simulation Procedure

---

**Data:** User  $u$ ,  $M_u$ ,  $TL_u$ ,  $Out_u$

- 1 Split  $M_u$  in two lists, one for ON sessions ( $on_u$ ) and other for OFF sessions ( $off_u$ )
- 2 **foreach** session  $s \in on_u$  **do**
- 3      $t_1 \leftarrow$  time when  $s$  begins
- 4      $t_2 \leftarrow$  time when the last OFF session in  $off_u$  that precedes  $s$  begins
- 5      $TL \leftarrow$  list of tweets in  $TL_u$  posted after  $t_1$  and before  $t_2$
- 6      $IT \leftarrow$  list of Interaction Tweets in  $s$
- 7     **foreach**  $m \in IT$  **do**
- 8         Update  $TL$  with messages posted before  $m$
- 9          $TL' \leftarrow$  Reorganize( $TL$ )
- 10         Search the origin of  $m$  in  $TL'$
- 11         Return the position of the origin of  $m$  in  $TL'$
- 12     **end**
- 13 **end**

---

retweet) we look for the original tweet (the replied or retweeted one) in the timeline of  $u$ . However, we look only in the most recent tweets, those received since the beginning of the last OFF session up to the time of the actual interaction (Lines 5 and 8). Before this search (Line 10), we perform one of the reorganizations algorithms described in Section 5 (Line 9). If the replied (retweeted) tweet is found in the timeline, its position is returned. It is important remark that not necessarily the tweet will be found in the timeline because it can be a reply or retweet of a non-followed user.

With this procedure we can compare our methodology with the normal Twitter timeline. It is only necessary compute the fraction of replied or retweeted tweets in the first  $p$  positions of the timeline when the reorganization method (Line 9) is one of the Section 5 or the simple chronological order.

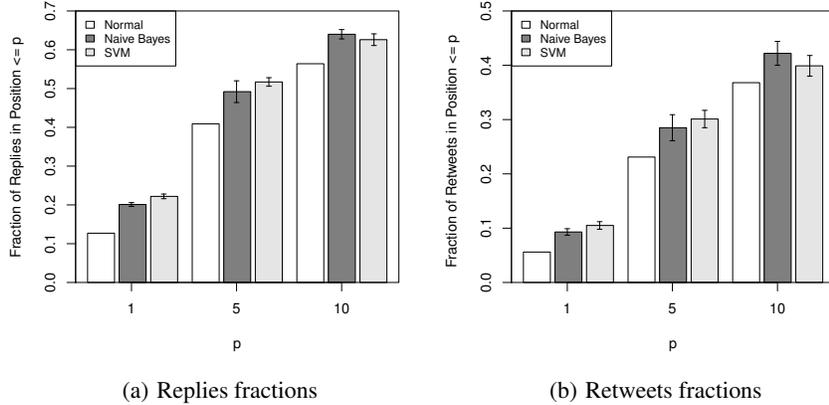
In order to have a good estimate of these values it is important to perform this simulations for a large number of users. The ideal scenario would be repeating it for all users in  $U$ . However, due to the computational cost, we need to use a sample. We extracted a random sample  $S$  from  $U$  with 10,000 users. In this sample we have 2.25 million tweets, of which 540 thousand are replies and 62 thousand retweets. Associated with this sample we have approximately 500,000 followed users whose were responsible for over 200 million received tweets.

Since we are working with samples, it is important guarantee that we have enough observations to have statistical confidence in the results. So we must be sure that the number of replies and retweets aforementioned are greater than a minimum threshold. According to Cochran [12] an approximation for a pessimist value in this case, with 95% of confidence and absolute error smaller than 0.005, is 38,416, showing that our sample is representative<sup>1</sup>.

After this step we divided this sample in five subsamples with 2000 users each and performed a cross validation procedure with them. We retained one subsample for training and used the others to test the machine learning algorithms. We performed this procedure 5 times. In each one, we used a different sample for training and the remaining for tests. For each test phase we computed the fraction of replies and retweets in the top- $p$  position of the timeline for  $p = 1$ ,  $p = 5$  and  $p = 10$ . Finally, we averaged each one of these

---

<sup>1</sup>The same argument can be used for the samples in Section 4. But, using 90% of confidence and an absolute error smaller than 0.01, which are still reasonable.



**Figure 7: Fraction of Replies/Retweets in the first  $p$  positions of the timeline. Errors bars are 95% confidence intervals.**

quantities using the results of all combinations of training and test subsamples.

Each machine learning algorithm has different training methodologies. For the NB, the training phase consists basically in estimating the parameters for the probabilities models presented in Section 5.2. For the SVM classifier we extracted a labeled trace of tweets and their attributes used for the classification. To perform this, we analysed our dataset and for each reply (retweet) we noted the data related to the replied (retweeted) tweet and selected other in the user’s timeline randomly that were not replied neither retweeted. After that we used this trace to train the classifier. For each sample we trained the classifier with information of replies, retweets and a third composed of the union of both. It is important remark that for the SVM classifier we used the library *Libsvm* [10] as base for simulations. In the training phase we used a tool available in that library to find the best parameters for the model.

## 6.2 Algorithm Evaluation

In this section we present the results obtained with the algorithms proposed in Section 5. Figure 7 presents the fraction of replies and retweets that are related with the first  $p$  tweets in the user timeline. These results were obtained when we trained the algorithms with information of replies and retweets. We proceeded in this way because this combination gave the best results.

We can see that both, NB and SVM algorithms improved significantly the fraction of replied tweets in the first positions of the timeline. Moreover, we have that NB and SVM have almost the same results (statistically equivalent).

In general, with this strategy we have good results in terms of presenting the most important tweets first for both types of interaction, where improvements of more than 50%, 20% and 10% for  $p$  equals to 1, 5 and 10 respectively were achieved.

## 6.3 $T_{OFF}$ parameter Impact

All results presented in the previous section use the parameter  $T_{OFF}$  equals to  $10^4$ s in the ON/OFF model presented in Section 5.1. But, is this value the most appropriate for all kind of Twitter users and situations? To answer this question we conducted one experiment where we varied  $T_{OFF}$  for sixteen different values in the range  $10^3$ s to  $10^5$ s. In this experiment we trained the algorithms with one sub sample (also used in previous section) with information of replies and retweets together and tested (executed the simulation procedure) in other one.

Table 2 shows the results of this experiment. To have a better

visualization we present only the minimum and maximum fractions among the sixteen values. We can see that for both algorithms (NB and SVM) the variation (difference between the maximum and minimum) is small, indicating that we can choose different values for  $T_{OFF}$  without a significant impact on the results.

**Table 2: Fractions of replies and retweets for different  $T_{OFF}$ .**

|         |    | SVM  |      | NB   |      |      |
|---------|----|------|------|------|------|------|
|         |    | $p$  | Min. | Max. | Min. | Max. |
| Reply   | 1  | 0.21 | 0.22 | 0.18 | 0.20 |      |
|         | 5  | 0.50 | 0.54 | 0.47 | 0.51 |      |
|         | 10 | 0.62 | 0.67 | 0.61 | 0.65 |      |
| Retweet | 1  | 0.07 | 0.08 | 0.07 | 0.08 |      |
|         | 5  | 0.23 | 0.28 | 0.23 | 0.27 |      |
|         | 10 | 0.32 | 0.38 | 0.34 | 0.39 |      |

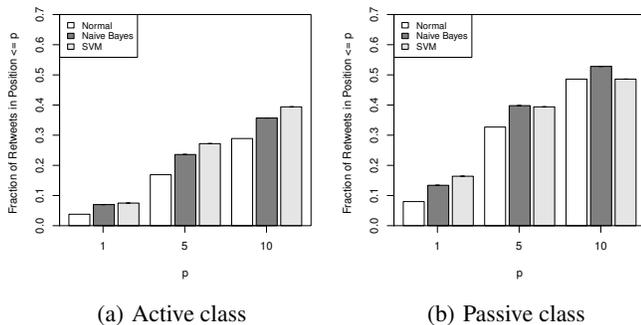
## 6.4 Active and Passive users

The objective of this section is to show that our algorithm is robust to work in classes of users with different patterns of activity. To that end, we conducted the same simulation procedure presented in last sections in two different classes of  $U$ : the first one is composed of active users, those that spend more time ON, and the second one of passive users, who spend less time ON. To divide these classes we defined the variable  $R_{ON}(u)$  as the fraction of time that user  $u$  is ON between his/her first and last tweets. After that we computed  $R_{ON}(u)$  for all users in  $U$  and considered the 20% of the users with the higher  $R_{ON}(u)$  as the active class and the remaining as the passive class.

We extracted a sample with 2000 users of each class, and used them in the simulation in order to investigate how our algorithms work in both cases. It is important to observe that we trained the algorithms with the same trace obtained of one subsample used in Section 6.2.

Figure 8 shows the results of this experiment. It is important remark that the same conclusions hold for replies, but we do not present the related figures due to space restriction. The first interesting point is that passive users tend to interact more with messages in the top of the timeline than active users. This is possible explained by the fact that active users spend more time ON and in this way, they spend more time looking at their timelines and interacting with tweets far from the top. Passive users tend to have more contact with tweets in top.

The second point is that improvements were achieved for both classes, especially for active users, where NB and SVM gave good



**Figure 8: Fraction of retweets in the first  $p$  positions of the timeline for active and passive classes.**

results. For the passive class, NB and SVM also worked well, except for SVM when  $p = 10$  in this last case.

## 7. CONCLUSIONS

In this paper we addressed the problem of understanding how users in Twitter interact with their timelines. Through an extensive characterization study we showed the importance of this problem and presented a set of relevant characteristics to deal with it. We showed that in general users prefer to interact with newer tweets, with users that they had previously interacted and with users with a lower sending rate (those that do not fulfill their timelines). Moreover, we showed that some basic textual characteristics such as, message size and the presence of hashtags, mentions and URLs affect interactions patterns, but in a different way, when we consider replies or retweets.

Motivated by these characterizations we proposed an algorithm to change the presentation order of the Twitter's timeline. Our methodology is based on two machine learning techniques, which showed significant improvements through a trace-driven simulation. Our main contribution is achieved when we used information of replies and retweets to train the classifiers. In this case we had good rates of replies and retweets in the top- $p$  positions of the timeline. Moreover, training the algorithms with information of a random sample over all users, we showed that our methodology was capable of working well for users in classes with distinct patterns of activity, showing the robustness of our method.

This is an important contribution once it represents another option of interface for the Twitter users, which can be specially interesting for mobile devices with a small screen. In addition, it is important remark that we used three simple attributes which are easy to compute. This fact makes our approach even more interesting for these devices once, in general, they have energy and memory constraints. In fact, we are currently building a Twitter client for iOS that implements our tweet-ordering algorithms, which we will distribute as open-source.

## 8. ACKNOWLEDGEMENTS

This research was funded by the Brazilian National Institute of Science and Technology for the Web (MCT/CNPq/INCT grant number 573871/2008-6) and by the authors' individual grants from CNPq, FAPEMIG and National Science Foundation (grants CNS-0905565, CNS-1018266, CNS-1012910 and CNS-1117039).

## 9. REFERENCES

- [1] Twitter blog. <http://blog.twitter.com/>.
- [2] P. André, M. S. Bernstein, and K. Luther. Who Gives A Tweet? Evaluating Microblog Content Value. In *Proceedings of CSCW 2012*, Feb. 2012.
- [3] M. Arlitt. Characterizing web user sessions. *SIGMETRICS Performance Evaluation Review*, 28(2):50–63, 2000.
- [4] E. Bakshy, J. M. Hofman, W. A. Madson, and D. J. Watts. Identifying 'Influencers' on Twitter. In *Web Search and Data Mining*, 2011.
- [5] A.-L. Bárábási. The origin of bursts and heavy tails in humans dynamics. *Nature*, 435:207, 2005.
- [6] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting spammers on twitter. In *Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, 2010.
- [7] M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi. Eddi: interactive topic-based browsing of social status streams. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, UIST '10, pages 303–312, New York, NY, USA, 2010. ACM.
- [8] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. The Socialbot Network: When Bots Socialize for Fame and Money. In *Annual Computer Security Applications Conference (ACSAC)*, Austin, TX, December 2010.
- [9] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi. Measuring User Influence in Twitter: The Million Follower Fallacy. In *Int'l AAAI Conference on Weblogs and Social Media (ICWSM)*, 2010.
- [10] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.
- [11] M. Choudhury, H. Sundaram, A. John, and D. Seligmann. What makes conversations interesting? themes, participants and consequences of conversations in online social media. In *World Wide Web Conference (WWW)*, pages 331–340, 2009.
- [12] W. G. Cochran. *Sampling Techniques*, 3rd Edition. John Wiley, 1977.
- [13] S. Counts and K. Fisher. Taking It All In? Attention in Microblog Consumption. In *AAAI Int'l Conference on Weblogs and Social Media (ICWSM)*, July 2011.
- [14] S. Ghosh, B. Viswanath, F. Kooti, N. K. Sharma, K. Gautam, F. Benevenuto, N. Ganguly, and K. P. Gummadi. Understanding and Combating Link Farming in the Twitter Social Network. In *Proceedings of WWW 2012*, April 2012.
- [15] T. Hastie, R. Tibshirani, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*. New York: Springer-Verlag, 2009.
- [16] C.-W. Hsu and C.-J. Lin. A simple decomposition method for support vector machines. *IEEE Transactions on Neural Networks*, 12:291–314, 1999.
- [17] B. A. Huberman, D. M. Romero, and F. Wu. Social networks that matter: Twitter under the microscope. *ArXiv e-prints*, Dec. 2008.
- [18] A. Java, X. Song, T. Finin, and B. Tseng. Why We Twitter: Understanding Microblogging Usage and Communities. *Proceedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop 2007*, August 2007.
- [19] B. Krishnamurthy, P. Gill, and M. Arlitt. A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, WOSN '08, pages 19–24, New York, NY, USA, 2008. ACM.
- [20] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Int'l World Wide Web Conference (WWW)*, pages 591–600, 2010.
- [21] D. Menascé, V. Almeida, R. Fonseca, and M. Mendes. A methodology for workload characterization of e-commerce sites. In *ACM conference on Electronic commerce (EC)*, 1999.
- [22] A. Oke and R. Bunt. Hierarchical workload characterization for a busy web server. In *Int'l Conference on Computer Performance Evaluation, Modelling Techniques and Tools (TOOLS)*, 2002.
- [23] T. Rodrigues, F. Benevenuto, M. Cha, K. P. Gummadi, and V. Almeida. On word-of-mouth based discovery of the web. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2011.
- [24] B. Suh, L. Hong, P. Piroli, and E. H. Chi. Want to be retweeted? Large scale analytics on factors impacting retweet in Twitter network. In *Proceedings of the IEEE Second International Conference on Social Computing (SocialCom)*, pages 177–184, 2010.
- [25] J. Weng, E.-P. Lim, J. Jiang, and Q. He. TwitterRank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM.