Predicting information diffusion on Twitter – Analysis of predictive features

Thi Bich Ngoc Hoang[a,b], Josiane Mothe[a,*]

[a] Université de Toulouse and IRIT, UMR5505 CNRS, France
[b] University of Economics, The University of Danang, Viet Nam

A B S T R A C T

Information propagation on online social network focuses much attention in various domains as varied as politics, fact checking, or marketing. Modeling information diffusion in such growing communication media is crucial in order both to understand information propagation and to better control it. Our research aims at predicting whether a post is going to be forwarded or not. Moreover, we aim at predicting how much it is going to be diffused. Our model is based on three types of features: user-based, time-based and content-based. Using three collections corresponding to a total of about 16 millions of tweets, we show that our model improves of about 5% *F*-measure compared to the state of the art, both when predicting if a tweet is going to be re-tweeted and when predicting how popular it will be. *F*-measure in our model is between 70% and 82%, depending on the collection. We also show that some features we introduced are very important to predict retweetability such as the numbers of followers and number of communities that a user belongs to. Our contribution in this paper is twofold: firstly we defined new features to represent tweets in order to predict their possible propagation; secondly we evaluate the model we built on top of both features from the literature and features we defined on three collections and show the usefulness of our features in the prediction.

Keywords:
Information diffusion
Predicting information propagation
Microblogs
Tweet propagation

1. Introduction

Online social networks are more and more popular as information channels. For example, smartinsights.com reports a penetration rate of about 89% for FaceBook and 32% for Twitter (US Internet users) for a total of 1871 million of active FaceBook users and 317 million twitter users in January 2017. Modeling information diffusion in such growing communication media is crucial in order both to understand information propagation and to better control it. Indeed, some studies have investigated the impact of social media in the recent elections both in US or in France, focusing mostly on fake news and their propagation on social media. The authors in [1] have collected 115 pro-Trump fake stories shared on Facebook for a total of 30 millions times while 41 pro-Clinton fake stories were shared a total of 7.6 million times. Since a high percent of voters use social media (35% of people 18–29 years old, according to Pew Research Center[1]), the hug number of share make fake stories successfully reach voters. Other examples could be found in marketing [2]. This illustrates the importance of understanding and predicting social media posts diffusion.

Our paper focuses on the prediction of information propagation on online social media. More precisely, we study two related questions: (1) Is it possible to predict whether a post (in our case a tweet) is going to be propagated (or re-tweeted)? and (2) Can the level of propagation be modeled and thus can we predict the level of propagation of a new post?

We answer these research questions by considering a model that we train on a subset of tweets and test on new tweets. Our model is based on three types of features: user-based, time-based and content-based. While some features come from previous work in the domain of tweet diffusion [3], we also introduce new features and evaluate the added value of these new features for both to predict whether a tweet is going to be retweeted or not and to predict the level of the propagation.

* Corresponding author at: Université de Toulouse and IRIT, UMR5505 CNRS, France.
E-mail addresses: thi-bich-ngoc.hoang@irit.fr (T.B.N. Hoang), josiane.mothe@irit.fr (J. Mothe).

https://doi.org/10.1016/j.jocs.2017.10.010

[1] http://www.journalism.org/2016/02/04/the-2016-presidential-campaign-a-news-event-thats-hard-to-miss/.

The rest of the paper is organized as follows: Section 2 presents the related work. Section 3 describes the features we used, giving a specific focus on the new features that we developed as well as the predictive model. Section 4 evaluates the model. Finally, Section 5 is the discussions and conclusion.

2. Related work

Information diffusion have attracted a number of researchers' attention in recent years. Several pieces of work have made efforts to study the prediction of information propagation in social network.

Suh et al. identified a number of features that may correlate with the number of retweets of a given tweet. They evaluated the correlation considering a large-scale analysis on 74 million tweets. They showed that the numbers of followers, followees, and ages of the account have a very strong relationship with the retweet number. The larger the number of the followers and followees of the sender is, the more likely his tweets get retweeted is. In addition, tweets posted by "senior users", who registered more than 300 days before writing, get a higher number of retweets than the average. On the contrary, the presence of hashtag or URL in a tweet does not highly correlates with the number of retweets. Suh et al. reported that 20.8% of retweets only contain hashtags while 28.4% of retweets contain URL. They also found that the number of past tweets has little or no relationship with the average number of daily tweets or with the retweet rate; the number of tweets that are favorited by users seem not to impact the retweetability since only 8.7% of retweets are written by authors with more than 100 favorited items [3]. In our work, we consider all the features proposed by Suh et al. including the presence of hashtags and URL in the tweet content, the number of followers, followees, number of tweets that a user has liked, total of past tweets and ages of the user's account [3]. We also add several new features including user-based, time-based, and content-based features.

Kwa et al. studied the relationship between the number of a user's followers and the popularity of his tweets for a collection of 106 million tweets. The authors constructed retweet trees and examined tree temporal and spatial characteristics. They showed that people only retweets from a small number of people and only a subset of a users followers actually retweet. In addition, users with less than 1000 followers tend to have the same average number of retweets for their posts [4]. Similarly, Remy et al. studied the impact of the number of followers of users on the capacity to propagate their message. Interestingly, they showed that the impact of users with a lot of followers is not statistically greater than users with a few followers [5]. This features is also considered in our work to analyze its impact on the retweetability.

Hong et al. casted the problem of predicting the popularity of tweets into binary classification and multi-class classification. They used logistic regression as a classifier considering the message content, meta data and structural properties of the users' social graph features on a 10,612,601-tweet collection. However, in their paper, Hong et al. did not describe the features they used explicitly. They achieved 0.60 *F*-measure for binary classification (recall 0.44 and precision 0.99). With regard to multi-classes classification, Hong et al. achieved good accuracy only for the smallest and largest categories: class-0 (not retweet) and class-3 (retweet number greater than 10,000). On the other hand, they got very low accuracy in the two other classes: 0.15 on class-1 (retweet number less than 100) and 0.43 on class-2 (retweet number less than 10,000) [6]. Our idea of classifying tweets into classes is similar to Hongs'. In the evaluation section of our paper (Section 4), we show that using Random forest as the machine learning algorithm and several new features we introduced, recall and *F*-measure can be improved for binary classification. We also improve the *F*-measure for class-1 and class-2 which are supposed to be more challenging classes since most of the tweets are in these two classes.

Hu et al. proposed an approach for predicting the short-term popularity of viral topics based on time series forecasting. They used historical popularity data of a given topic and considered three types of features: previous-popularity-based, user-comment-based and network-structure-based. They showed that the popularity is relatively dynamic and changeable for burst topics and historical popularity can still have an impact on later popularity for non-burst topics [7]. Xiong et al. characterized information propagation on Twitter by considering the topic of the tweet. They supposed that users select the topic that they are most interested in and then retweet. The more topics a user participates in, the less the user will turn attention to a new topic. Xiong et al. also supposed the inhibition between topics is important to user's decision. As a result, by using more than 20,000 tweets to train the model, they found that individual decision making mainly depends on the topic itself [8]. In the work presented in this paper, we did not consider the topic of the tweet but instead we added several content features which users may use to enhance the tweet content such as checking if the tweet contains location name, company name, TV show, picture or video.

Yang et al. also studied the retweet process on social network. From their first observation on twitter data, they found that almost 25.5% of the tweets posted by users are actually retweeted from their friends' statuses. From that, they proposed a semi-supervised framework on a factor graph model to predict Twitter user's retweeting behaviors. The features of the users' history preferences, messages content and information of the trace were considered but are not explicitly described in their paper. In the experiments, Yang et al. reported *F*-measure of 0.33 on the prediction, outperforming the L1-regularized logistic regression method. However their method did not outperform the Support Vector Machine baseline in terms of recall [9]. With similar interest, Zhang et al. addressed the problem of how users' behaviors are influenced by friends in their ego network. They first tested whether the influence locality exists in the microblog network and whether it significantly influences user's retweet behavior. They found that the fraction of active users (retweeted a message) with two active neighbors (followees who have retweeted the same message) is about double compared to the fraction of active users with only one active neighbors. They also showed that, although the probability a user retweets a message is positively correlated with the number of active neighbors, it is negatively correlated with the number of connected circles that are formed by those neighbors [10]. We did not consider the influence of followers' retweeting behavior on friends in our work since the datasets we use do not contain information of users' followers; this could be an interesting feature for our future work.

3. Predicting information diffusion: features and model

The model in itself is based on machine learning; with this respect it is similar to Hong's, which used machine learning techniques to predict the popularity of messages as measured by the number of future retweets [6] (see Section 2).

Using machine learning implies that (1) each tweet is represented by a set of features (2) a training set is used in order to learn the model before the model is used on the test set or new tweets.

3.1. Tweet representation

We hypothesize that both the tweet content and the user who wrote it have an impact on tweet diffusion. To decide on possible useful features to represent tweets, we manually analyzed about

Table 1
Features used to predict retweet rate of a given tweet. Features with a$^+$ correspond to Suh et al. features [3] while the other features correspond to one important contribution of this paper.

Features	Description	Data type	
User-based	1. Total_of_tweets$^+$	Total of past tweets that the user has posted in the timeline	#Numeric
	2. No_of_followers$^+$	Number of people who follow the user	#Numeric
	3. No_of_followees$^+$	Number of people the user follows	#Numeric
	4. Age_of_account$^+$	Number of days since the user account has been created	#Numeric
	5. No_of_favourite$^+$	Number of tweets the user has liked in the timeline	#Numeric
	6. No_groups_user_belongs	Number of groups that the user belongs to	#Numeric
	7. Aver_favou_per_day	Average of likes that the user has made per day	#Numeric
	8. Aver_tweets_per_day	Average of tweets that the user has posted per day	#Numeric
	9. User_name_len	The length of the user's name	#Numeric
Time-based	10. Is_post_at_hol	The tweet is created on public holiday	Boolean
	11. Is_posted_at_noon	The tweet is created from 11 a.m.–13 p.m.	Boolean
	12. Is_posted_at_eve	The tweet is created from 6 p.m.–9 p.m.	Boolean
	13. Is_post_at_wee	The tweet is created at weekend	Boolean
Content-based	14. Contain_location	The tweet contains a location name	Boolean
	15. Contain_org	The tweet contains an organization name	Boolean
	16. Contain_tvshow	The tweet contains a television show name	Boolean
	17. Sentiment_level	The tweet is classified into sentiment levels	{positive, negative, objective}
	18. Contain_video	The tweet contains a video	Boolean
	19. Contain_picture	The tweet contains a picture	Boolean
	20. Contain_upper	The tweet contains upper words	Boolean
	21. Contain_number	The tweet contains number	Boolean
	22. Contain_excl	The tweet contains an exclamation mark	Boolean
	23. Contain_rt_term	The tweet contains RT term	Boolean
	24. Con_user_mentioned	The tweet mentions a user name	Boolean
	25. Contain_rt_sugges	The tweet contains one of the retweet suggestion term: Pls RT, please retweet, RT for . . .	Boolean
	26. Contain_URL$^+$	The tweet contains an URL	Boolean
	27. Contain_hashtag$^+$	The tweet contains a hashtag	Boolean
	28. Opt_length	Length of the content is between 70 and 100 characters	Boolean
	29. Len_of_text	Length of the content	# Numeric

500 tweets from the Sandy collection [11]. The idea was to detect clues that could be useful to predicted retweet or/and the retweet rate. We also relied on large scale analytics of factors affecting retweetability [3] to enrich the tweet representation.

Finally, in our model, tweets are represented by user-based, time-based and content-based. There are a total of 29 features. The features along with their short description are presented in Table 1.

Shu et al. mentioned that some features highly correlate with retweet rate such as number of followers, number of followees, age of the user's account while other features have slight impact only on this rate such as the presence of URL and hashtag. Moreover, the total number of past tweets and the number of tweets that are favorited by the user seem to have little or no relationship with the retweet number [3]. We reuse all these features in our model. Those features are marked with a$^+$ in Table 1. The other features are features that we defined and correspond to one main contribution of this paper.

3.1.1. User-based features

We hypothesize that a person who highly interacts with other people will in turn receive corresponding attention. Thus we take into account the interaction between the user who sends the tweet and social network. We first reused the features that are related to the retweet number mentioned in [3]:

Total_of_tweets: the total tweets that the user has posted in the past.
No_of_followers: the number of people who follow the user.
No_of_followees: the number of people the user follows.
Age_of_user: the number of days since the user account has been created until the day the tweet was collected.
No_of_favourite: the total number of tweets the user has liked in the timeline.

In addition, we added some new features:

No_groups_user_belongs the number of groups or communities that the user belongs to.
Aver_favou_per_day: this features is calculated by dividing *No_of_favourite* by *Age_of_user*.
Aver_tweets_per_day: this features is calculated by dividing *Total_of_tweets* by *Age_of_user*.
User_name_len: the length of the user's name.

All the features from this category are numeric values. These features are extracted and calculated from the fields a tweet is composed of when collected using Twitter API.

3.1.2. Time-based features

We hypothesize that a majority of retweets are written shortly after the tweet is posted and thus that a tweet posted in 'free hours' is more likely to receive more retweets. The time-base features that consider the time the tweet is generated, include:

1 *Is_post_at_hol:* we check if the tweet is posted during holidays using the Holiday python library.[2] We first consider the public holiday of user's location during the time of collecting the datasets (as available in Section 4.1). If the user does not mention any location in her or his profile, we check the tweet posting time with holidays of all 23 countries which is included in the Holiday python library such as United States, United Kingdom, Spain, Germany and others.
2 *Is_posted_at_noon*: we check whether the tweet is posted at noon from 11 a.m. to 13 p.m. or not.

[2] https://pypi.python.org/pypi/holidays.

3 *Is_posted_at_eve:* we check whether the tweet is posted in the early evening from 5 p.m. to 9 p.m. or not.
4 *Is_post_at_wee:* we check whether the tweet is posted at the weekend or not.

Each of these checks corresponds to a boolean feature in the tweet representation.

3.1.3. Content-based features

We added several new content-based features considering the content of the message such as named entity, sentiment level, media attachment, content enhancement, content size and others.

Named entity: A tweet that mentions a specific location name makes it more attractive [12] and may lead to retweetability. For example, the tweet: *"Tonight's moonrise over the #statueofliberty in New York City."* got 1,200 retweets. Also, a TV show or a business company included in a tweet makes it more popular. 4,600 people have retweeted the post: *"Heres a look at our #PrimeDay sneak peek of #TheGrandTour Season 2"*. We used Ritter's named entity extraction tool [13] to check if the tweet contains a location name (*Contain_location*), an organization name (*Contain_org*) or a TV show reference (*Contain_tvshow*). We suppose that information about well-known named entities included in the tweet will get much attention and will be shared more. These features are boolean values. We distinguish between sentiment level, media attachment, Content enhancement, and content size. These features are presented in the following sub-sections.

Sentiment level: We hypothesize that in special events such as epidemics or promotion campaigns, extremely positive or negative tweets are normally used to express hot and updated news and these tweets are more prone to be retweeted. For example, the tweet about the death toll from a hurricane in Haiti *"The death toll in Haiti from Hurricane Matthew is 339. That's what environmental racism looks like. #BlackLivesMatter"* got more attention as 1,500 retweets were posted in a short time. We thus defined a new feature to capture the sentiment of tweets that we called Sentiment_level. We used a "scikit-learn" machine learning library[3] to classify tweets into positive, negative or neutral sentiment. We trained the model on the training dataset including 6,030 annotated sentiment tweets provided by Semval-2013 international workshop on Semantic Evaluation, Sentiment analysis on Twitter task[4] and on 10,600 shorten annotated sentiment movie reviews.[5] From our experiments, among classifiers, stochastic gradient descent SGD classifier gave the highest accuracy on the training set thus this classifier was used to extract sentiment features in the three collections of tweets described in Section 4.1. we kept three possible values of this sentiment feature: positive, negative or objective.

Media attachment: Twitter users often attach media sources to make their tweets more lively and more attractive. A picture attached in a message *"When you're finally home alone and u could be yourself"* probably contributed this tweet got 2,231 retweets. We therefore defined features related to attached items. More specifically, we check if the tweet contains a picture *(Contain_picture)* or a video *(Contain_video)*. These two features are boolean values.

Content enhancement: We take into account some features that can enhance retweetability such as the fact the tweet contains an upper word (*Contain_upper*), a number (*Contain_number*), an exclamation mark (*Contain_excl*), a 'RT' term (*Contain_rt_term*) or mentions a user name (*Con_user_mentioned*). These features that we defined are boolean values.

Table 2
The number of tweets and their distribution on the Sandy, FirstWeek and SecondWeek datasets used to evaluate our predictive model.

	Sandy	FirstWeek	SecondWeek
# of tweets	2,119,854	8,009,112	8,171,080
# of tweets which are not retweeted	1,156,223	4,025,157	4,058,066
# of unique tweets which are retweeted	204,232	2,017,979	2,080,962

We also consider some retweet suggestion terms may be effective in asking people to retweet (*Contain_rt_suggest*) including: 'please retweet', 'pls rt', 'retweet if', 'rt if', 'retweet to', 'rt to', ' rt!', 'retweet for', 'rt for', 'retweet' e.g. *"For every retweet this gets, Pedigree will donate one bowl of dog food to dogs in need! #tweetforbowls"*. We thus check if tweets contain one of the above terms. This feature is a boolean value.

Besides, we reapply two boolean features from [3] which check if the tweet contains a URL (*Contain_URL*) or a hashtag (*Contain_hashtags*).

Content size: We consider the length of the tweet content which is limited to 140 characters (*Len_of_text*). We suppose that the ideal length of a message should be in between 70 and 100 characters so that there is space for people to put comments in addition to the content that they want to retweet (*Opt_length*). The former feature is numeric while the later feature is boolean.

3.2. Processing time

The feature extraction process was implemented on the Osirim-IRIT platform[6] with 1 CPU 1.6 GHz, and 64 GB of RAM.

For each dataset, we extract the features from the tweets that are not retweeted and from unique tweets which are retweeted. Since a tweet may be retweeted several times, it can be stored repeatedly in datasets. We thus only consider the original tweet one time with the latest 'number of retweets'. It took one week to extract features for the FirstWeek dataset and one week for the SecondWeek dataset but just few days for the Sandy dataset because of fewer number of tweets as presented in Table 2.

3.3. Machine learning model

There are several commonly used machine learning algorithms that could have been used for our purpose. We used different machine learning algorithms such as Naive Baiyes (NB), Support Vector Machine (SMO) and Random Forest (RF) implemented on Java Weka library.[7]

For each collection, we used 10-folds cross validation. We also formed an experiment that implement transfer learning: we trained the model on one collection and tested it on a different collection.

Among these classifiers, RF consistently achieved the best results which are reported in the next session.

4. Evaluation of the model

4.1. Data and evaluation framework

We conducted experiments and evaluated our model on three collections which were collected from Twitter APIs: Sandy, FirstWeek and SecondWeek datasets.

[3] http://scikit-learn.org/stable/.
[4] https://www.cs.york.ac.uk/semeval-2013/task2/index.html.
[5] https://pythonprogramming.net/new-data-set-training-nltk-tutorial/.
[6] IRIT, UMR5505 CNRS, France.
[7] http://weka.sourceforge.net/doc.stable/.

The first dataset has initially been used by Tamine et al. [11] collected from 29th October 2012 to 31st October 2012 using the 3 keywords "sandy", "hurricane" and "storm" while the second and the third datasets were 1 percent of tweets collected during the first week and second week of January 2017 by IRIT, France[8] within a spam detection project [14].

Each tweet in these datasets is composed of pieces of information regarding a twitter's post such as the unique identifier (id), the content of the tweet, the time this tweet was created, the author of this tweet and others. We used the value of the 'retweet_count' field which specifies the numbers of times a tweet has been retweeted to classify tweets in the predictive model (Section 4.2).

Table 2 reports the number of tweets and their distribution in the three datasets.

Baseline: The baseline model we report in this section uses all the Suh's features [3]. We compare it with the model that considers all the features we presented in Table 1 including the one we defined in this paper.

4.2. Experiments and results

4.2.1. Binary classification

Since there is a huge difference between the number of tweets in class-0 (tweets that are not retweeted) and tweets in class-1 (tweets that are retweeted), we balanced these numbers during the classification process. There are several ways to deal with imbalanced data such as resampling the dataset, generating synthetic samples or penalizing models.[9] We chose to divide each dataset into several sub-sets. The tweets from class-1 are all kept whatever the sub-set is while the tweets from class-0 are divided into sub-sets so that the number of tweets in class-0 is approximately the same as the number of tweets in class-1 for each sub-set. More specifically, the sub-sets are built as follows:

- **Sandy dataset**. The tweets from class-0 were divided into five parts. Each sub-set included the entire class-1 (204,232 tweets) and one part of class-0 (about 231,245 tweets). We had thus five sub-sets for which we consider the average results when reporting them in Table 4.
- **FirstWeek dataset**. The class-0 was divided into two parts. Each sub-set included the whole class-1 (2,017,979 tweets) and one part of class-0 (about 2,012,579 tweets). We had thus two sub-sets for which we consider the average results when reporting them in Table 4.
- **SecondWeek dataset**. Like for FirstWeek dataset, the class-0 was divided into two parts. Each sub-set included the whole class-1 (2,080,962 tweets) and one part of class-0 (2,029,033 tweets). As in the previous case, we had two sub-sets for which we consider the average results when reporting them in Table 4.

Table 4 reports the *F*-measure of the binary classification (a tweet is predicted to be retweeted or not) on Sandy, FirstWeek and SecondWeek datasets. * indicates statistically significant differences by Student's *t*-test. For each dataset, we report the average of *F*-measure over the sub-sets.

As it can be seen in Table 4, we significantly improve the *F*-measure of the binary classification on average and on every class compared to the baseline for all datasets.

On average, we achieved the *F*-measure of 0.70 for the Sandy dataset while this number is 0.65 for the baseline; it corresponds to an improvement of 5%, statistically significant. For both the FirstWeek and the SecondWeek datasets, the *F*-measure is improved from 0.78 to 0.82 which corresponds to an improvement of 4% statistically significant. When training the model on the FirstWeek and testing on the SecondWeek dataset, we obtained the *F*-measure of 0.82 compared to 0.8 for the baseline 2% of improvement, statistically significant.

Interestingly, our model achieves higher performance on class-1 (tweets that are retweeted) than on class-0 (tweet that are not retweeted) even if the number of tweets in class-1 is smaller than the number of tweets in class-0. For the Sandy dataset, the *F*-measure on class-1 is increased by 0.06 while it increases by 0.04 on class-0 compared to the baseline. When the model is trained on the FirstWeek and tested on the SecondWeek dataset, the *F*-measure is improved by 0.03 on class-1 but just 0.01 on class-0.

Table 3
Classes distribution of Sandy, FirstWeek and SecondWeek datasets used for muti-class classification. Class-0 corresponds to tweets that are not retweeted at all; class-1 represents tweets that are retweeted less than 100 times; class-2 represents tweets that are retweeted less than 10,000 times; class-3 represents tweets that are retweeted more than 10,000 times.

	Sandy	FirstWeek	SecondWeek
Class-0	1,156,223	4,025,157	4,058,066
Class-1	202,397	1,675,859	1,727,666
Class-2	1832	327,381	339,328
Class-3	3	14,739	13,905

4.2.2. Multi-class classification

To predict the volume of retweets that a particular message will receive in the future, we divided the messages into four different classes like Hong et al. did [6]: class-0 corresponds to tweets that are not retweeted at all, class-1 represents tweets that are retweeted less than 100 times, class-2 represents tweets that are retweeted less than 10,000 times, and finally class-3 represents tweets that are retweeted more than 10,000 times.

Table 3 presents the class distribution of Sandy, FirstWeek and SecondWeek collections.

As can be seen in Table 3, the number of tweets in classes is are very imbalanced. To solve this problem we combine two steps:

- **Step 1.** Generating synthetic samples to randomly sample the attributes from instances in classes-2 and class-3 using Synthetic Minority Over-sampling Technique (SMOTE). This algorithm selects some similar instances (using a distance measure) and perturbs an instance, one attribute at a time by a random amount within the difference to the neighboring instances [15]. We configure SMOTE implemented on java Weka library to oversample class-2 and class-3 as follow: setNearestNeighbors = 5 and setPercentage = 100. As a result, the number of tweets in class-2 and class-3 were doubled.
- **Step 2.** We divided each dataset into numbers of sub-sets like for binary classification. The tweets of class-1, class-2 (after SMOTE) and class-3 (after SMOTE) were kept the same for all sub-sets while the tweets of class-0 were divided into sub-sets so that the number of tweets in class-0 was approximately equal to the number of tweets in class-1, specifically:
  - **Sandy dataset**. The class-O tweets were divided into five parts. Each sub-set included the whole class-1, class-2 (after SMOTE) and class-3 (after SMOTE) with a total of 206,067 tweets and one part of class-O tweets including about 231,245 tweets. We had thus five sub-sets for which we consider the average results when reporting them in Table 5.
  - **FirstWeek**. The class-O was divided into two parts. Each sub-set included the whole class-1, class-2 (after SMOTE) and class-3 (after SMOTE) with a total of 2,360,099 tweets and one part of class-O including about 2,012,579 tweets. We had thus five sub-

[8] IRIT, URM CNRS 5505 Université de Toulouse, France.
[9] http://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/.

Table 4
F-measure of the binary classification using Random Forest on the three datasets..

	Sandy			FirstWeek			SecondWeek			Training on FirstWeek, testing on SecondWeek		
	Cl-0	Cl-1	Aver.	Cl-0	Cl-1	Aver.	Cl-0	Cl-1	Aver.	Cl-0	Cl-1	Aver.
Baseline	0.69	0.61	0.65	0.79	0.77	0.78	0.79	0.77	0.78	0.86	0.67	0.80
Our method	0.73	0.67	**0.70***	0.83	0.81	**0.82***	0.82	0.81	**0.82***	0.87	0.70	**0.82***

* Statistically significant differences when using Student's *t*-test.

Table 5
F-measure of the multi-class classification using Random Forest on the three datasets.

Datasets	Classes	Baseline (*F*-measure)	Our method (*F*-measure)
Sandy	Cl-0	0.69	0.73
	Cl-1	0.60	0.66
	Cl-2	0.53	0.55
	Cl-3	0.81	0.93
	Aver.	0.65	**0.70***
FirstWeek	Cl-0	0.79	0.82
	Cl-1	0.64	0.70
	Cl-2	0.73	0.74
	Cl-3	0.58	0.57
	Aver.	0.72	**0.76***
SecondWeek	Cl-0	0.79	0.82
	Cl-1	0.65	0.74
	Cl-2	0.73	0.74
	Cl-3	0.57	0.57
	Aver.	0.72	**0.76***
Training on FirstWeek trained, testing on Second Week	Cl-0	0.85	0.86
	Cl-1	0.51	0.55
	Cl-2	0.58	0.65
	Cl-3	0.45	0.55
	Aver.	0.73	**0.75***

* Statistically significant differences when using Student's *t*-test.

sets for which we consider the average results when reporting them in Table 5.

– **SecondWeek**. The class-O was divided into two parts. Each subset included the whole class-1, class-2 (after SMOTE) and class-3 (after SMOTE) with a total of 2,434,132 tweets and one part of class-O including about 2,029,033 tweets. As in the previous case, we had two sub-sets for which we consider the average results when reporting them in Table 5.

These divisions do not completely guarantee the exact balance among classes, but reduce the importance of the majority class(es).

Table 5 presents the results of multi-class classification on three datasets in terms of averaged *F*-measure over sub-sets, * indicates statistically significant differences by Student-test.

Similarly to binary classification, our method significantly improves the *F*-measure of the multi-classes classification on average and on every class compared to the baseline for all three datasets.

On average, comparing to the baseline, we improve the *F*-measure by 5% for the Sandy dataset (from 0.65 to 0.7), 4% for both the FirstWeek and SecondWeek dataset (from 0.72 to 0.76) and 2% when training the model on the FirstWeek and testing on the SecondWeek datasets (from 0.73 to 0.75). All these improvements are significant different.

On every class of all the three datasets, our methods improves the *F*-measure compared to the baseline but with different performances. We achieved high *F*-measure on class-0, class-1 and class-2 (from 0.70 to 0.82) but lower *F*-measure on class-3 (0.57) for the FirstWeek and SecondWeek datasets. This may be caused by the large difference of the numbers of tweets per classes. The number of tweets in class-1 is about five time the number of tweets in class-2 and more than one hundred times the number of tweets in class-3.

Compared to the FirstWeek and the SecondWeek datasets, we achieved lower *F*-measure for the Sandy dataset. The *F*-measures on class-0, class-1 and class-2 are 0.73, 0.66 and 0.55 respectively. However, we got very high *F*-measure on class-3 as it is 0.93. Since the number of tweets on class 3 is extremely small compared to thousand or hundreds of thousand in other classes, the similarity between the tweets from class-3 may have lead to the high performance of the classification for this class.

4.2.3. Most important features

Our predictive model uses 29 features of which we have proposed 22 in this paper as a contribution. Some of these features are more useful than others to predict retweet numbers. We evaluated the importance of each feature by measuring the so called Infor-gain attribute evaluator using Ranker search method in Weka. This tool calculates the relative weight of each feature in the model. The results are presented in the next sections.

4.2.3.1. Binary classification. The best five features when classifying tweets in binary classes are as follows (numbers in brackets corresponds to the weight; the higher the value is, the more important the feature is for the model):

- **Sandy dataset:** No_of_followers$^+$ (0.118), No_groups_user_belongs (0.100), Is_posted_at_eve (0.077), Is_posted_at_noon (0.044), No_of_followees$^+$ (0.033).
- **First_Week dataset:** No_of_followers$^+$ (0.227), No_groups_user_belongs (0.113), Is_post_at_hol (0.072), No_of_followees$^+$ (0.047), No_of_favourite$^+$ (0.041).
- **Second_week dataset:** No_of_followers$^+$ (0.237), No_groups_user_belongs (0.130), No_of_followees$^+$ (0.051), No_of_favourite$^+$ (0.043), Contain_picture (0.041).

We found that two features we reapply from Suh et al. (number of followers and followees) are consistently in the top five features. This result matches with their finding that number of followers and followees have a very strong relationship with the retweetability. On the contrary, number of tweets that the user has liked in his timeline was found to have very little impact on the retweet number by Suh et al. [3] while it is one of the best five features on our Firstweek and Secondweek datasets.

One important result is that one of the new features we defined, number of groups or communities that the user belongs to (No_groups_user_belongs), is the second best features over for the three datasets. The results also show our time-base features play an important role in predicting whether the tweet is retweeted or not. The retweetability of a given tweet on two over three collections is affected by the time posting features: in the evening (Is_posted_at_eve) and at noon (Is_posted_at_noon) or during holiday (Is_post_at_hol).

Contain_picture is the most important content-based feature in the five top features of SecondWeek dataset while this feature is the sixth best in the FistWeek dataset and sixteenth best in Sandy dataset. The low rank of Contain_picure in the Sandy dataset may

be caused by the very small number of tweets containing pictures since most of tweets in this dataset are about Sandy hurricane.

Apart from the above features, the next important features on three datasets with different weight are: Aver_tweets_per_day, Total_of_tweets$^+$, Len_of_text, Aver_favour_per_day, Contain_hashtag $^+$, User_name_len, Contain_URL$^+$, Sentiment_level, Con_user_mentioned, Contain_rt_suggestion.

4.2.3.2. Multi-class classification. Similarly to binary classification, two features from the literature No_of_followers$^+$, No_of_followees$^+$ and one of features that we defined (No_groups_user_belongs) are consistently in the best five features.

More precisely, the best five features when classifying tweets in multi-class classification are as follow:

- **Sandy dataset:** No_of_followers$^+$ (0.141), No_groups_user_belongs (0.119), Is_posted_at_eve (0.077), Is_posted_at_noon (0.045), No_of_followees$^+$ (0.038).
- **First_Week dataset:** No_of_followers$^+$ (0.329), No_groups_user_belongs (0.228), Len_of_text (0.213), No_of_followees$^+$ (0.131), Age_of_account$^+$ (0.115).
- **Second_week dataset:** No_of_followers$^+$ (0.372), No_groups_user_belongs (0.331), Len_of_text (0.262), No_of_followees$^+$ (0.150), Age_of_account$^+$ (0.125).

On the contrary, while the number of tweets that the user has liked in his timeline (No_of_favourite) is very important for binary classification, it is not so important in multi-class classification. Instead, the tweet length (Len_of_text) is significant while it was not for binary classification. Indeed it is the third best feature in both the FirstWeek and the SecondWeek datasets. Our result for the Age_of_account feature matches with Suh's finding when they showed that it has a significant relationship with retweet rate. In both the FirstWeek and SecondWeek datasets, Age_of_account is the fifth best feature with the weights 0.115 for the FirstWeek dataset and 0.125 for the SecondWeek dataset.

When considering Sandy dataset, the order of the best five features in multi-class classification is the same as in binary classification, although the weights are little higher for all the features. The top five features in multi-class classification for the FirstWeek and the SecondWeek datasets are similar; but relatively different from those for binary classification. The Is_post_at_hol, Contain_picture and No_of_favourite $^+$ features are significant in binary classification but not in multi-class classification.

Apart from the above features, the next important features on the three datasets are: Aver_tweets_per_day, Aver_favour_per_day, Total_of_tweets$^+$, Contain_picture, No_of_favourite$^+$, Contain_hashtag$^+$, User_name_len, Contain_URL$^+$, Sentiment_level, and Con_user_mentioned.

4.2.4. Correlations between features

To evaluate if the new features we defined are dependent from existing features and independent each others, we calculated the correlations between features. We applied the Principle Component evaluator using Ranker search method implemented on Weka. We obtained a correlation matrix which measures the degree of association between features for each dataset. We also used R programing language to visualize the correlations.

Fig. 1 presents the correlation matrix between all the features for the Sandy dataset. The higher the correlations, the larger and bolder the circles. We did not display the three obtained visualizations since they are very similar in shape.

The first important point is that there are a few correlations that are significant; most of them are weak correlations. As it can be seen in Fig. 1, and this holds also for the two other datasets, most of the features are independent from each other. Indeed, most of the

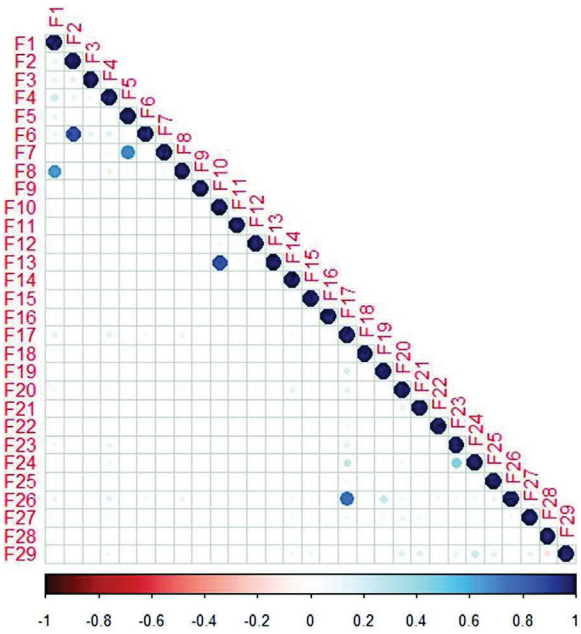

Fig. 1. The correlation between features in the Sandy dataset. The large and bold circles represent high correlations. The features are in the same order as in Table 1.

correlation values are between −0.2 and 0.2 for the three datasets. The highest correlations in each dataset are as follow:

- **Sandy dataset:** *No_groups_user_belongs correlates with No_of_followers$^+$(0.86)*; Is_post_at_week correlates with Is_post_at_hol (0.86); Sentiment_level correlates with Contain_URL$^+$ (0.75); Aver_favou_per_day correlates with No_of_favourite$^+$ (0.68); Aver_tweets_per_day correlates with Total_of_tweets$^+$ (0.65);
- **FirstWeek dataset:** *No_groups_user_belongs correlates with No_of_followers$^+$(0.74)*; Sentiment_level correlates with Con_user_mentioned (0.53); Contain_picture correlates with Contain_URL$^+$ (0.5); Aver_favou_per_day correlates with Aver_tweets_per_day (0.45);
- **SecondWeek dataset:** *No_groups_user_belongs correlates with No_of_followers$^+$(0.84)*; Sentiment_level correlates with Con_user_mentioned (0.52); Contain_picture correlates with Contain_URL$^+$ (0.49); Is_post_at_week correlates with Is_post_at_hol (-0.33).

As it can be seen, the correlations for the FirstWeek and the SecondWeek datasets are very similar to each other but slightly different from the Sandy dataset. The only significant correlation that exists across the three datasets is between No_groups_user_belongs (a feature that we defined) and No_of_followers$^+$ (a feature from the literature). Apart from this, the other significant correlations are between existing features and some features that we defined but that have little weights in the predictive model and thus which are not important for the model.

For example, in the Sandy dataset, Sentiment_level (which correlates with Contain_URL$^+$) got 0.0009 importance weight while the weight of the Aver_favou_per_day feature (correlates with No_of_favourite$^+$) is 0.003. In addition, Aver_tweets_per_day which correlates with Total_of_tweets$^+$ is also a weak feature in our model.

In the FirstWeek and the SecondWeek datasets, most of correlations are between our less important features. The Sentiment_level feature, which got 0.0019 importance weight, correlates with Con_user_mentioned feature, which got 0.013. Besides, other features which correlate to each other such as Aver_favou_per_day,

Aver_tweets_per_day, Is_post_at_week and Is_post_at_hol are not important for the predictive model. The Contain_picture feature which correlates with Contain_URL$^+$ is important in binary classification for the SecondWeek dataset but it is not important whatever the classification is for the FirstWeek dataset.

To conclude, there is very few meaningful correlations between the features in the three datasets; most of the correlation values being in between −0.2 and +0.2. When considering the correlations that are statistically significant between the features that we defined in this paper and features from the literature are not important for the predictive model (low weights). Some of the features that we developed in this paper are both significant for the predictive models (main features) and do not correlate with existing features from the literature. This is the case for Is_posted_at_noon, Is_posted_at_eve, Is_post_at_hol, and Len_of_text. Moreover, the results presented in Section 4.2 show that the combination of our features and existing features significantly improves the performance of the predictive information-diffusion model.

5. Discussions and conclusion

In this paper, we address the problem of predicting whether a given tweet will be retweet or not. We also adress the challenge of predicting the volume of retweets that a certain tweet will receive. We developed new features to represent tweets and also reused some features from the literature. We applied the machine learning model using random forest classifier. The new features we proposed are of three types: user-based, time-based and content-based features. We show that, our model improves by about 5% *F*-measure compared to the state of art (statistically significant) for both types of prediction when evaluating our model on three collections of total of about 16 millions.

There are some features that are more important than others. We show that the number of followers, followees, and the number of groups that the user belongs to, are the most important features for both types of prediction and consistently across the datasets; the third feature being suggested in this paper. In addition, the time-based features we developed to check if a tweet is posted at noon, in the evening, at weekend or during holiday also strongly correlate with the retweetability. These two new features do not correlate with features from the literature.

Indeed, we also analyzed the correlations between features in the three datasets. Most of features are independent from each others. The few features of ours that correlate with existing features, have generally low weights when analyzing their impact for the predictive models.

In addition, the results presented in Section 4.2 show that the combination of the features we defined and existing features significantly improves the performance of the predictive model.

There are several points that could be considered for future work. The three datasets we used to evaluate our predicted model were collected during a rather short time: the Sandy dataset was collected during a three days period while the Firstweek and Secondweek were collected in one week. Thus, it could be interesting to analyze further the impact of tweet posting time on retweetability when considering datasets collected in longer period of time.

For future work, we will apply the document vector model (Doc2vec) as a new feature [16] which will be trained on the FistWeek dataset and to infer vectors for tweets on the SecondWeek dataset. Our hypothesis is that adding these vector features to our model will bring interesting result We hypothesis that if Doc2Vec was learn from topics, event and stories from the training set, it would infer better vectors for the testing set. We also would like to classify topics of the tweet into different categories such as music, movie, fashion and business. We believe that some people are more interested in some topics than on other. Tweets about one's favorite topics are more likely to be retweeted by him or her. Finally, a track could be to analyze the influence when a follower retweets a tweet on one of his/her friends.

References

[1] H. Allcott, M. Gentzkow, Social Media and Fake News in the 2016 Election, Tech. rep., National Bureau of Economic Research, 2017.

[2] T. Hennig-Thurau, C. Wiertz, F. Feldhaus, Does twitter matter? The impact of microblogging word of mouth on consumers adoption of new movies, J. Acad. Market. Sci. 43 (3) (2015) 375–394.

[3] B. Suh, L. Hong, P. Pirolli, E.H. Chi, Want to be retweeted? Large scale analytics on factors impacting retweet in twitter network, in: 2010 IEEE Second International Conference on Social Computing (SocialCom), IEEE, 2010, pp. 177–184.

[4] H. Kwak, C. Lee, H. Park, S. Moon, What is twitter, a social network or a news media? in: Proceedings of the 19th International Conference on World Wide Web, ACM, 2010, pp. 591–600.

[5] C. Remy, N. Pervin, F. Toriumi, H. Takeda, Information diffusion on twitter: everyone has its chance, but all chances are not equal, in: 2013 International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2013, pp. 483–490.

[6] L. Hong, O. Dan, B.D. Davison, Predicting popular messages in twitter, in: Proceedings of the 20th International Conference Companion on World Wide Web, ACM, 2011, pp. 57–58.

[7] Y. Hu, C. Hu, S. Fu, P. Shi, B. Ning, Predicting the popularity of viral topics based on time series forecasting Neurocomputing 210 (2016) 55–65.

[8] F. Xiong, Y. Liu, Z.-j. Zhang, J. Zhu, Y. Zhang, An information diffusion model based on retweeting mechanism for online social media, Phys. Lett. A 376 (30) (2012) 2103–2108.

[9] Z. Yang, J. Guo, K. Cai, J. Tang, J. Li, L. Zhang, Z. Su, Understanding retweeting behaviors in social networks, in: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ACM, 2010, pp. 1633–1636.

[10] J. Zhang, B. Liu, J. Tang, T. Chen, J. Li, Social influence locality for modeling retweeting behaviors, IJCAI, vol. 13 (2013) 2761–2767.

[11] L. Tamine, L. Soulier, L. Ben Jabeur, F. Amblard, C. Hanachi, G. Hubert, C. Roth, Social media-based collaborative information access: analysis of online crisis-related twitter conversations, in: Proceedings of the 27th ACM Conference on Hypertext and Social Media, ACM, 2016, pp. 159–168.

[12] J. Lingad, S. Karimi, J. Yin, Location extraction from disaster-related microblogs, in: Proceedings of the 22nd International Conference on World Wide Web, ACM, 2013, pp. 1017–1020.

[13] A. Ritter, S. Clark, O. Etzioni, et al., Named entity recognition in tweets: an experimental study, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2011, pp. 1524–1534.

[14] M. Washha, A. Qaroush, F. Sedes, Leveraging time for spammers detection on twitter, in: Proceedings of the 8th International Conference on Management of Digital EcoSystems, ACM, 2016, pp. 109–116.

[15] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, Smote: synthetic minority over-sampling technique, J. Artif. Intell. Res. 16 (2002) 321–357.

[16] Q.V. Le, T. Mikolov, Distributed representations of sentences and documents, ICML, vol. 14 (2014) 1188–1196.

Thi-Bich-Ngoc Hoang She is a lecturer at University of Economics, the University of Danang, Vietnam. In 2011, she received her master degree in information systems development in Hogeschool van Arnhem en Nijmegen, The Netherlands. Currently, she is a PhD student in computer science at Université de Toulouse, France. Her research interest is mainly in social networks analysis and information systems modeling.

Josiane Mothe She is Professor in computer science at the ESPE (teacher training school) Université de Toulouse since 2002. She is a specialist in information retrieval, data mining and big data. From 2012 to 2015, she has been leading the Information System team of the French IRIT lab, CNRS Unit. From 2004 to 2014, she was the editor in chief for Europe and Africa of the international Information Retrieval Journal, (Springer). Since 2011 she is co-responsible of the "MicroBlog contextualization task" at the CLEF evaluation forum. Eleven PhD students were supervised to successful completion by her. She is now leading the FabSpace 2.0 project, an open innovation network for geodata-driven innovation, funded under H2020 Research and Innovation program.