Fake News Detection Using Time Series and User Features Classification

Marialaura Previti[1], Victor Rodriguez-Fernandez[2(✉)], David Camacho[3], Vincenza Carchiolo[4], and Michele Malgeri[1]

[1] Dip. di Ingegneria Elettrica, Elettronica e Informatica (DIEEI), Università degli Studi di Catania, Catania, Italy
marialaura.previti@unict.it, michele.malgeri@dieei.unict.it
[2] Universidad Autónoma de Madrid, Madrid, Spain
victor.rodriguezf@uam.es
[3] Departamento de Sistemas Informaticos, Technical University of Madrid, Madrid, Spain
david.camacho@upm.es
[4] Dip. di Matematica e Informatica (DMI), Università degli Studi di Catania, Catania, Italy
vincenza.carchiolo@unict.it

Abstract. In a scenario where more and more individuals use online social network platforms as an instrument to propagate news without any control, it is necessary to design and implement new methods and techniques that guarantee the veracity of the disseminated news. In this paper, we propose a method to classify true and false news, commonly known as fake news, which exploits time series-based features extracted from the evolution of news, and features from the users involved in the news spreading. Applying our methodology over a real Twitter dataset of precategorized true and false news, we have obtained an accuracy of 84.61% in 10-fold cross-validation, and proved experimentally that all the selected features are relevant for this classification task.

Keywords: Fake news detection · Random forest classification · Time series features · User information features

1 Introduction

In the last decades, due to the increasing amount of time spent by a large part of world population interacting through Online Social Networks (OSNs), more and more individuals tend to seek out and consume news from OSNs, instead of using traditional mass media, such as newspapers and television. There are several reasons for this change of behaviors:

– in cases where modern communication technologies are unavailable, OSNs users, not belonging to any category of journalists, can provide news in real time through these platforms (*gatewatching* [1]), often helping press agencies to collect information in order to provide breaking news. For example, they allow identifying the outbreaks of infectious diseases in real time [2], detecting the spread of seasonal epidemics, such as influenza, in order to organize

P. A. Castillo et al. (Eds.): EvoApplications 2020, LNCS 12104, pp. 339–353, 2020.
https://doi.org/10.1007/978-3-030-43722-0_22

containment measures [3–5], detecting and tracking discussion communities on vaccination [6], or detecting information about natural disasters in order to manage rescue and promptly warn affected populations [7–9];
- the access to OSNs is fast, always available and less expensive than traditional mass media and help users to select topics they are interested in;
- through OSNs, interactions with news (through like and repost) and with other users interested in the topic of posts (through exchange of comments) are possible. Instead, in traditional mass media communication is unidirectional.

Albeit these advantages, the quality of news in social media is often low, due to the lack of authoritative sources that check their veracity. This helps malicious users to propagate false news over the network. Moreover, the possibility of choice among many sources of information conducts some groups of individuals to seek out sources of information which reinforce their pre-existing point of view about a specific topic, generating the *echo chambers phenomenon* [10]. In this context, the credibility of news spreaders and the frequency of exposure to the same piece of news play an important role.

The intent of false (or *fake*) news is to manipulate news in order to add false information with purpose to create a damage (political, economical or reputational among others) to an adverse counterpart or to perform surreptitious advertising to their products or personal characteristics in order to gain an advantage in comparison with competitors. In order to help big companies and political parties in this intent, often behind OSNs accounts there are not physical users, but bots and cyborg users whose sole purpose is to propagate an idea reaching as many users as possible in a specific category of users [11].

A short and complete definition of fake news was provided by Allcott and Gentzkow [12], they established that fake news is *news intentionally and verifiably false that could mislead readers*. Typically, fake news is spread over OSNs in form of rumors, i.e., groups of posts related to the same topic propagate by different users and reposted several times.

In such a wide scenario with so many facets, it is necessary to find strategies to stop the proliferation of fake news on OSNs and, for this purpose, the early identification in true or false is a fundamental step. We propose a method to classify rumors exploiting their temporal evolution and information about user involved in rumor spreading.

The initial decision to exploit the temporal evolution of the rumors in classification was taken thanks to the studies done in [13], because the authors have shown that the temporal evolution of the true and false news are different, but at this starting point we added the information about users involved in each rumor, because from a similar study conducted on two OSNs dataset [14], the only use of time series produces poor results in the early stages of propagation that improve when more temporal information is acquired, while the information about users (e.g., user followers, followees and engagement) contains previous clues on the diffusion capacities of the involved individuals, making our method effective from the initial stages of propagation. Moreover, but not less important, our method

does not require the use of tweet texts, so it requires reduced computing capacity to calculate the used features and reduced storage capacity, because it is not necessary to extract and store the entire tweets but only some basic information, in fact the elaborations of this paper were carried out with a simple mid-range notebook and a subset of tweets collected in the starting phase of propagation, while maintaining accuracy results consistent with the results currently existing in the literature.

The rest of the paper has been structured as follows. Next section provides a short introduction to the current state of the art in the area of fake news detection. In Sect. 3, we describe our methodology that consists in the classification of features extracted from time series representing temporal evolution of rumors (rumors are collections of tweets about the same topic and in the context of OSNs can be qualified as news) and information related to users involved in each rumor. In Sect. 4, we apply this methodology on a real dataset and perform an experimental analysis for each feature. Finally, in Sect. 5 the main conclusions and some future lines of work are outlined.

2 Related Work

In this section, we report only the most relevant attempts to identify fake news on OSNs, a simple taxonomy of those approaches can be organized in *content-based approaches* and *context-based approaches*:

The **content-based** approaches aim at finding clues in texts and images of OSNs posts useful to differentiate false and true news, hence the majority of them are linguistic approaches.

For example, in [15] the authors focused on mining particular linguistic cues as patterns of pronouns, conjunctions and words that arouse in readers negative emotions, while in [16], authors use Rhetorical Structure Theory (RST platform) to represent rhetorical relations among the words in the text and to extract style-based features of news by mapping the frequencies of rhetorical relations to a vector space, in [17] authors use Linguistic Inquiry and Word Count (LIWC platform) to extract the lexicons falling into psycholinguistic categories, exploiting a large set of words that represent psycholinguistic processes, summary categories, and part-of-speech categories. These two classifiers, usually used for long texts, were applied to Twitter messages by [18] obtaining accuracy respectively of 60% and 72%.

Considering the enormous amount of work needed to create good training sets, in the last years, many researchers use debunking agencies (i.e., organizations whose goal is to check the contents of news assigning them a "value of veracity") to construct their dataset of true/false news and, after the text preprocessing, they use traditional classifiers. For example, Ferreira and Vlachos [19] developed a stance classification approach based on multiclass logistic regression, using features extracted from the article headline and the claim, achieving an accuracy of 73% on their dataset Emergent, Wu et al. [20] proposed a graph-kernel based hybrid SVM classifier which captured the high-order propagation

patterns in addition to semantic features such as topics and sentiments reaching a classification accuracy of 88% in false rumor early detection on Sina Weibo dataset.

The visual-based approaches try to identify fake images that are intentionally created or obtained by the capture of specific characteristics in larger images. These techniques are often used in combination with other ones in order to obtain good results, for example, in the case of faking image related to hurricane Sandy, Gupta et al. [21] performed a classification, not only of news contents, but also of users who posted fake tweets.

On the other hand, there are approaches based on **social context** that include relevant user social engagements in their analysis, capturing this information from various perspectives. These approaches can be: stance-based [22,23], that utilize users' viewpoints (e.g., provided in form of like and repost) from relevant post contents to infer the veracity of original news articles, or propagation-based, that reason about the interrelations of relevant social media posts to predict news credibility.

For instance, in [24], authors proposed a credibility analysis approach based on PageRank-like credibility propagation on a multi-typed network consisting of events, tweets and users. For each interaction, their algorithm updated the event credibility scores, and they proved that their approach is 14% more accurate than the decision tree classifier approach on the same dataset.

Another method to automatically assess credibility of news propagated through Twitter was provided by Castillo et al. [25], that exploited a TwitterMonitor to identify trend topics and analyzed the collected tweets to discard conversations and unsure tweets, keeping only tweets labeled as news, and considering message, user, topic and propagation features to assign a value of credibility to each topic. The value obtained was put into several classifiers obtaining, in the best case, 86% of accuracy. This platform was reused in [18] on BuzzFeed ad PoliFact datasets (two datasets of tweets verified by debunking organizations) obtaining an accuracy of 80% and 79.6% respectively.

Finally, Ma et al. [14] proposed an approach to capture the variations of social context features during the propagation of post over time exploiting Dynamic Time Series Structure model that use the features of rumor's life cycle. They obtained an accuracy that range from 78% to 89% after 25 h from the start of propagation on a Twitter dataset and from 77% to 86% in 49 h from the start of propagation on a Sina Weibo dataset. In this model the accuracy in the initial phase of propagation is lower respect to the end because it lacks of sufficient variation of social context.

3 Extraction of User Information and Time Series Features for Fake News Detection on Twitter

This work was inspired by the work of Vosoughi et al. [13]. They performed an investigation on differential diffusion of verified true and false news stories propagated on Twitter between 2006 and 2017 and, after a deep analysis of

126 thousand stories twitted by 3 million of individuals they concluded that: *"Falsehood diffuses significantly farther, faster, deeper, and more broadly than the truth in all categories of information."*.

According to these conclusions, our approach consists on extracting features from the *temporal evolution* of each rumor and features from the users involved in each rumor propagation, then, after a classifier pre-training phase, evaluate the quality of each feature by Gini-importance method and use only the most important ones for the specific dataset in the final classification through random forest classifier.

To perform this kind of analysis we need a dataset composed of several raw tweets belonging to specific rumors, whose veracity has already labelled as true or false in some way (e.g. by exploiting links to debunking website pages). In particular, we are interested in the following fields of each tweet:[1]

- `created_at` that is the datetime of the current tweet creation;
- `id_str`, that is the identification number of the current tweet;
- the following fields of user object of current tweet object:
  - `id_str`, that is the identification number of user who has posted the current tweet;
  - `friends_count`, that is number of followees;
  - `followers_count`, that is number of followers;
  - `listed_count`, that is the number of public lists a user is member of (e.g., when he receives a reply);
  - `favorites_count`, that is number of tweets the user has liked during his account's lifetime;
  - `statuses_count`, that is number of tweets and retweets issued by the user during his account's lifetime;
  - created_at, that is the datetime that the user account was created on Twitter;
- if the current tweet is a retweet, the following fields of original tweet object:
  - `created_at`, that is the datetime of the original tweet creation;
  - `id_str`, that is the identification number of the original tweet.

Below are detailed the two families of features that are proposed in this work for the characterization of news on Twitter. The combination of all of them comprises the feature set that will be used to feed the classification algorithm.

3.1 Extraction of Time Series-Based Features

Datetimes of original and subsequent tweets are used to construct the timeline of each rumor. We select the oldest tweet of each rumor as origin and calculate a timestamp for the remaining part of rumor tweets, as the difference (in seconds) between the tweet datetime and the origin. After this, we construct a time series for each rumor by computing how many tweets there are for each hour in the first 24 h, so each rumor time series is the sum of the time series of cascade that belong to rumor translate over the time axis, as explained in toy example of Fig. 1.

[1] The names of fields has been extracted from the Twitter developers documentation.

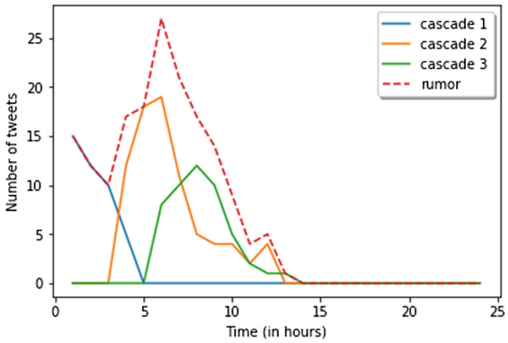

Fig. 1. Example of the construction of a rumor time series: there are 3 cascades (i.e. original tweets with all their retweet) with different starting time. The oldest tweet (belonging to the first cascade) marks the starting point of the rumor and the other 2 cascades are shifted on X axis respect to it. The sum of the 3 shifted curve points constitute the rumor curve.

The decision of taking into account just the first 24 h of propagation is based on some preliminary analyses on cascades of a subset of Vosoughi et al. dataset [26], and on the use of rumors in other works (e.g., [27]). These works confirm that the bulk of the news propagation happens in the first hours after carrying out of the event that generated news, and is drastically reduced in about a week. Therefore, it is useless to process the whole temporal evolution of news spreading in order to obtain a good classification. Moreover, the temporal evolution highlighted by these works led us to consider not seasonal time series equations to represent the temporal evolution of rumors, in fact, the final step consists in extracting numerical features from the time series of each rumor.

The collection of time series features has been performed through the `tsfeatures` tool (list of features for this type of analysis in [28]). We selected the most promising among the features proposed in [29] and [30], and after a pre-training of a random forest classifier, we computed the Gini-importance (i.e., the difference between a node's impurity and the weighted sum of the impurity measures of the two child nodes in tree [31]) to remove the unsuitable features for our dataset.

Thus, we consider our non-seasonal time series as $x_t = f_t + e_t$, where f_t is the smoothed trend component computed using the *smoother of Friedman* [32], and e_t is a remainder component. Based on this, the following features are extracted:

- Strength of trend, i.e., $1 - \frac{Var(e_t)}{Var(f_t+e_t)}$, where $Var(e_t)$ is the variation of e_t.
- Spike measures the *spikiness* of a time series, and it is computed as the variance of the leave-one-out variances of e_t.
- Linearity and curvature measures the linearity and curvature of a time series calculated based on the coefficients of an orthogonal quadratic regression.

- Autocorrelation function of e_t (e_acf1 and e_acf10), keeping the value of the first autocorrelation coefficient and the sum of the first ten squared autocorrelation coefficients.
- Shannon entropy, i.e., $-\int_{-\pi}^{\pi} f(x)logf(x)dx$ where $f(x)$ is an estimate of the spectral density of data.
- Stability and lumpiness are two features based on not overlapped tilled windows and means and variances are produced for all windows, the first one is the variance of means and the second one is the variance of variances.
- Crossing points, i.e. the number of time series crosses the mean line.
- Max level shift, max var shift and max Kullback-Leibler shift are features based on overlapping windows, they find respectively the sizes and the time indexes of the largest mean shift, the largest variance shift and the largest shift in Kullback-Leibler divergence between two consecutive windows.

3.2 Adding User Information to the Feature Set

Exploiting only the temporal evolution of the news is not enough to make a robust characterization and detection of their veracity, so, the feature set obtained from time series is extended with information about the users that participated in the rumor. More specifically, we calculate the average of user followers, followees and engagements for all the users involved in all the cascades of a rumor spread. The average of user followers and followees give us an idea of how connected the users of the rumor are within the Twitter network. Instead, the average of user engagement indicates how active the users were since the creation of their Twitter accounts, up to the time in which they tweeted or retweeted a piece of news. User engagement takes into account the number of tweets (T), retweets (Rt), replies (Rp) and favorites (F) of a user during the entire time he stayed on Twitter platform in days (D):

$$Eng = \frac{T + Rt + Rp + F}{D} \tag{1}$$

4 Experimentation

4.1 Dataset

For the experimental part of this paper, we used a partial Twitter dataset provided by Vosoughi, Roy and Aral [13]. They did not provide the *raw Twitter data* to comply with the Twitter policy on user privacy. Dataset contains the fields mentioned in the previous section for each tweet (properly anonymized), as well as, a rumor id and a cascade id, in order to identify the tweets belonging to the same rumor and cascade without the use of the tweet text (not provided by authors), and the veracity (true, false or mixed) of the news deriving from the analyzes carried out by 6 debunking agencies.

We removed tweets categorized as "mixed" to avoid uncertainty in the classification. Also, we split the tweets by rumor, in order to apply the method

mentioned in the previous section to construct the rumor time series. Isolated tweets, i.e., tweets that do not contribute to the propagation of any rumor, were also removed from the dataset. Another reduction of the dataset was performed by truncating each rumor after 24 h of propagation, by removing rumors with less than 30 tweets and with a duration of less than 10 h, due to these rumors do not give a big picture of news spreading. At the end of this preprocessing, our dataset was composed of 1998 rumors (1582 false, 406 true) with a total of 1,503,990 tweets. In order to obtain a balanced dataset, we used 406 true rumors and 406 false rumors, randomly selected.

4.2 Features Extraction and Classification Results

For the training phase, we applied a random forest classifier with 500 trees, the rest of the parameters were set to the default values given by the implementation of random forest used in the R package `mlr` [33].

Table 1. Gini importances. The features in bold are kept for classification. Considering the great difference between the higher and lower value in table, we used as threshold the arithmetic average between these two values in order to decide which fields keep.

TS feature	Importance	TS feature	Importance	Dataset information	Importance
trend	**27.3315**	stability	**23.5289**	av_user_followers	**30.14687**
spike	**30.12765**	crossing_points	12.71993	av_user_followees	**29.24244**
linearity	**31.46726**	max_level_shift	**26.58879**	av_user_engagement	**29.24244**
curvature	**28.02144**	time_level_shift	2.166012	category	9.128586
e_acf1	**24.31782**	max_var_shift	**29.12224**		
e_acf10	**26.51979**	time_var_shift	3.463651		
entropy	**25.62444**	max_kl_shift	**24.04622**		
lumpiness	**21.41172**	time_kl_shift	1.747805		

After training the model, we applied the Gini-importance technique that calculates the importance of each feature. As explained before, this is calculated as the decrease of impurity for classification. The results are shown in Table 1.

We kept only the fields that surpass the mean value between the obtained minimum and maximum values. Considering the low importance of the features `crossing_points`, `time_level_shift`, `time_var_shift`, `time_kl_shift` and `category`, we removed these columns from the dataset and rerun training and, performing the 10-fold cross-validation, we obtained $84.61 \pm 5.96\%$ of accuracy.

4.3 Accumulated Local Effect Plot Analysis

In order to interpret the predictions of the resulting model, we will apply recent model-agnostic methods from the field of eXplainable AI (XAI) [34,35]. The

advantage of these methods over model-specific ones is their flexibility, because they can be applied to any model, and provide different ways of representing the explanations.

First of all, we will analyze the interaction between features. If these features interact with each other, the prediction can not be expressed as the sum of the feature effects, because the effect of each feature is influenced by other ones, hence the way to estimate the interaction strength among features is to measure how much of the prediction variation depends on the interaction of the features.

This goal can be reached using H-statistic measurement [36]:

$$H_j^2 = \frac{\sum_{i=1}^{n} [f(x^{(i)}) - PD_j(x_j^{(i)}) - PD_{-j}(x_{-j}^{(i)})]}{\sum_{i=1}^{n} f^2(x^{(i)})} \tag{2}$$

where $f(x)$ is the prediction function, $PD(x_j)$ is the partial dependence function of feature j, $PD(x_{-j})$ is the partial dependence function of all features except j and n is the number of data points used for measurement.

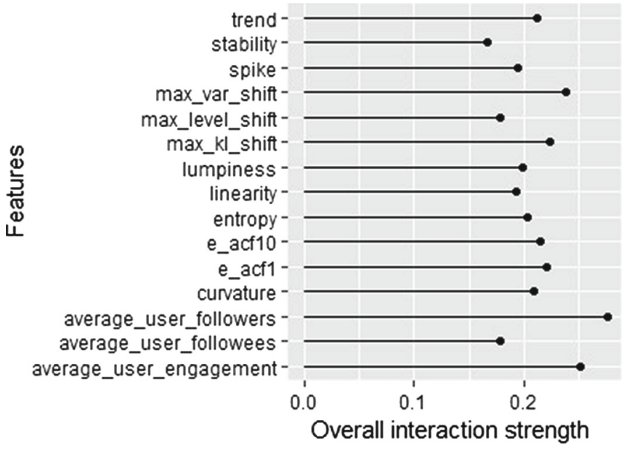

Fig. 2. Interaction strength of each feature versus the rest.

Figure 2 shows the application of this measure to each feature used in our dataset. All the features have weak interaction with other ones (ranging in [0.16, 0.28]). The average user followers has the highest relative interaction effect, followed by average user engagement, hence features deriving from time series are interacting with user features.

Finally, in order to interpret the resulting model, we analyze its Accumulated Local Effects (ALE) plot [37]. ALE is a common technique in the field of explainable machine learning. ALE plots are a faster and unbiased alternative to Partial Dependence Plots (PDP) and Individual Conditional Expectation (ICE),

since they overcome the problems of model interpretation when the features are correlated as in this case.

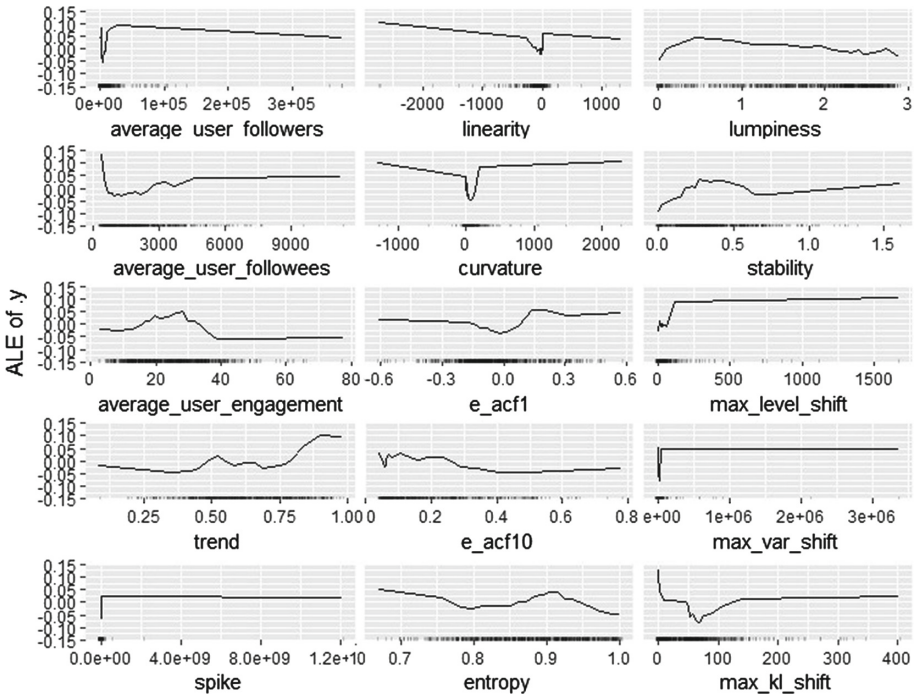

Fig. 3. Accumulated Local Effects (ALE) for each feature.

To estimate local effects, ALE method implementation divides the feature into many intervals and computes the differences in the predictions, that is the effect each feature has for each individual instance in a certain interval. All the effects of each interval are summed and after divided for the number of instances in each interval in order to obtain an average, the *local effect*. To accumulate the effects, the contributions of each interval are summed, giving the ALE value for a certain feature. More detail about the theory behind this estimation in [38].

In Fig. 3 the effect of each feature in the prediction of false news is shown. The corresponding charts for true news has been omitted because, in a binary classification, the curves are mirrored with respect to the X axis.

Looking at the effect of **user-based features** (*average user followers*, *average user followees* and *average user engagement*), we see that false news spreaders usually have a large number of followers, a low number of followees and a low user engagements. This probably means that their accounts were created ad hoc to reach a particular category of users and only become active when a particular idea must be propagated, remaining inactive for the rest of the time.

Looking at the ALE curves of **time series-based features**, we observe that, for values close to 0 of *linearity* and *curvature*, the prediction is often true, while it is false for the rest of values of these features. That indicates that, unlike false rumors, true news tend to have a constant diffusion. Considering the above mentioned formula for *strength of trend*, for high variations of the f_t component the rumor is classified as false, which hints that most of the false news have an higher variation of spreading over time with respect to true news. Finally, regarding the *max_level_shift* and *max_var_shift* we see that, except for low value of these two features, the classifier gives false news prediction. True rumors tend to have fewer shifts in the evolution of number of tweets than fake news.

Observing the whole figure, we can see that there are no features with high ALE value for all the value of each feature. This means that there is not a feature with predominant importance that alone can help to easily classify the news, but each feature contribute in the formation of the result, this means that removing one or more features the accuracy will decrease.

4.4 Tuning the Level of Aggregation of the Rumor Time Series

In the above experiment, the tweet counts needed for creating the rumor time series were calculated on an hourly basis. However, it is interesting to study whether the use of shorter levels of aggregation (also known as unit of analysis or sampling rate) can obtain better values of classification accuracy. Thus, we aggregated the tweets with a range of levels of aggregation, from 10 to 60 min, in 10-min increments and re-apply our methodology (Table 2).

Table 2. Average accuracy and standard deviation for different levels of aggregation of the rumor time series.

Level of aggregation	Average accuracy
10 min	$84.97 \pm 3.16\%$
20 min	$85.46 \pm 5.6\%$
30 min	$84.11 \pm 2.02\%$
40 min	$82.76 \pm 4.14\%$
50 min	$83.75 \pm 3.84\%$
60 min	$84.61 \pm 5.96\%$

4.5 Studying the Evolution of the Classification Accuracy over Time

Considering that the best result found in the previous experiment is that using a 20-min level of aggregation maximizes the accuracy on 24-h length rumor time series, we carried out a study of the evolution of the accuracy with this level of aggregation, in order to understand if the overall time of 24 h of spreading

is necessary to obtain a good classification, or instead, a smaller time can be considered without losing too much accuracy. This is really important because an early detection of false news is always most desirable that a higher classification accuracy in a later time, when fake news have already become viral.

Taking into account that some of the time series-based features (e.g., linearity and curvature) cannot be calculated if the time series are too short, we started our analysis with 21 points (which correspond, in 20-min steps, to 7 h of propagation after the publication of the first tweet about a rumor) and re-executed the proposed methodology with 10-fold cross-validation resampling strategy, just as in the previous experiments. The average and deviation of the accuracy values found after this experiment are reported in Fig. 4. There are not high variations in accuracy: the lowest values of accuracy in average is $82.02 \pm 4.06\%$ at 7 h and 20 min, the highest $86.95 \pm 2.9\%$ at 10 h. This shows that, by reducing the length of time series to 8 h instead of the initial 24, the accuracy remains over 82%, which is a fair value, specially if we take into account the earliness of that classification. This is due to we did not exploit just the information about the propagation, but also the information about the involved users. So, unlike similar approaches that use time series (e.g., [14]) and need hours before to reach a stable values of accuracy, we have quite constant results in classification.

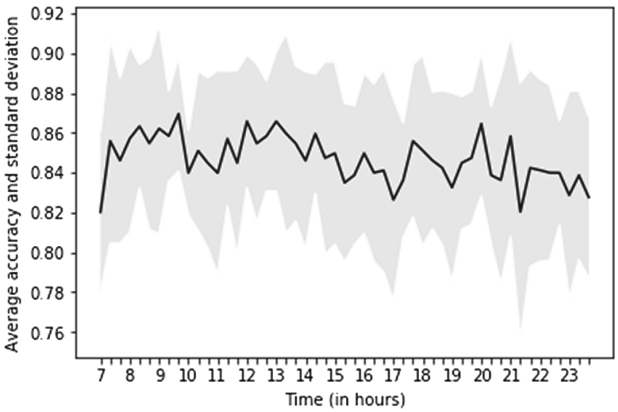

Fig. 4. Accuracy and standard deviation chart in 20 min time steps classifications.

5 Conclusions

In this work, we presented a method to classify true and false news spread on Twitter, exploiting both time series-based features and information about users involved in news spreading. We applied the feature set to a dataset of precategorized news, obtaining an accuracy in ten-fold cross-validation of 84.61% for 24 h time series with tweets sampling of 1 h. In order to obtain this result,

we studied the importance of each feature after a pre-training process, and we used only the most important ones to perform the best classification possible for our dataset. The relevance of these features was also proved by a posterior study about the effect of each feature in the classification. Finally, we re-performed the same methodology on time series of the same dataset but with different level of aggregation, in order to find the one that allow the highest accuracy result, and, in this case, we performed a study of the evolution of accuracy over the time and, considering that the accuracy vary between 82% and 87% in all the cases, we reached to the conclusion that time series can be truncated at 8 h with a low loss of classification accuracy.

As future work, an important extension of the proposed methodology lies in the combination of the current set of features with others derived from the tweet metadata (e.g., tags, geographic locations). This new extension will be compared against other techniques and algorithms currently used in the area of fake news detection using more datasets.

Acknowledgements. This work has been supported by several research grants: Spanish Ministry of Science and Education under TIN2014-56494-C4-4-P grant (DeepBio), European Union, under ISFP-POLICE ACTION: 823701-ISFP-2017-AG-RAD grant (YoungRes), and Comunidad Autónoma de Madrid under P2018/TCS-4566 grant (CYNAMON).

References

1. Bruns, A.: The active audience: transforming journalism from gatekeeping to gatewatching (2008)
2. Chunara, R., Andrews, J.R., Brownstein, J.S.: Social and news media enable estimation of epidemiological patterns early in the 2010 Haitian cholera outbreak. Am. J. Trop. Med. Hyg. **86**(1), 39–45 (2012)
3. Lee, K., Agrawal, A., Choudhary, A.: Real-time disease surveillance using twitter data: demonstration on flu and cancer. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1474–1477. ACM (2013)
4. Christakis, N.A., Fowler, J.H.: Social network sensors for early detection of contagious outbreaks. PLoS ONE **5**(9), e12948 (2010)
5. Schmidt, C.W.: Trending now: using social media to predict and track disease outbreaks. Environ. Health Perspect. **120**(1), a30 (2012)
6. Bello-Orgaz, G., Hernandez-Castro, J., Camacho, D.: Detecting discussion communities on vaccination in twitter. Future Gen. Comput. Syst. **66**, 125–136 (2017)
7. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes twitter users: real-time event detection by social sensors. In: Proceedings of the 19th International Conference on World Wide Web, pp. 851–860. ACM (2010)
8. Guy, M., Earle, P., Ostrum, C., Gruchalla, K., Horvath, S.: Integration and dissemination of citizen reported and seismically derived earthquake information via social network technologies. In: Advances in Intelligent Data Analysis IX, pp. 42–53 (2010)
9. Spence, P.R., Lachlan, K.A., Griffin, D.R.: Crisis communication, race, and natural disasters. J. Black Stud. **37**(4), 539–554 (2007)

10. Barberá, P., Jost, J.T., Nagler, J., Tucker, J.A., Bonneau, R.: Tweeting from left to right: is online political communication more than an echo chamber? Psychol. Sci. **26**(10), 1531–1542 (2015)
11. Varol, O., Ferrara, E., Davis, C.A., Menczer, F., Flammini, A.: Online human-bot interactions: detection, estimation, and characterization. In: Eleventh International AAAI Conference on Web and Social Media (2017)
12. Allcott, H., Gentzkow, M.: Social media and fake news in the 2016 election. J. Econ. Perspect. **31**(2), 211–236 (2017)
13. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
14. Ma, J., Gao, W., Wei, Z., Lu, Y., Wong, K.-F.: Detect rumors using time series of social context information on microblogging websites. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 1751–1754. ACM (2015)
15. Feng, V.W., Hirst, G.: Detecting deceptive opinions with profile compatibility. In: Proceedings of the Sixth International Joint Conference on Natural Language Processing, pp. 338–346 (2013)
16. Rubin, V.L., Lukoianova, T.: Truth and deception at the rhetorical structure level. J. Assoc. Inf. Sci. Technol. **66**(5), 905–917 (2015)
17. Pennebaker, J.W., Boyd, R.L., Jordan, K., Blackburn, K.: The development and psychometric properties of LIWC2015. Technical report (2015)
18. Shu, K., Wang, S., Liu, H.: Beyond news contents: the role of social context for fake news detection. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 312–320. ACM (2019)
19. Ferreira, W., Vlachos, A.: Emergent: a novel data-set for stance classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1163–1168 (2016)
20. Wu, K., Yang, S., Zhu, K.Q.: False rumors detection on Sina Weibo by propagation structures. In: 2015 IEEE 31st International Conference on Data Engineering, pp. 651–662. IEEE (2015)
21. Gupta, A., Lamba, H., Kumaraguru, P., Joshi, A.: Faking Sandy: characterizing and identifying fake images on twitter during Hurricane Sandy. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 729–736. ACM (2013)
22. Mohammad, S.M., Sobhani, P., Kiritchenko, S.: Stance and sentiment in tweets. ACM Trans. Internet Technol. (TOIT) **17**(3), 26 (2017)
23. Qazvinian, V., Rosengren, E., Radev, D.R., Mei, Q.: Rumor has it: identifying misinformation in microblogs. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1589–1599. Association for Computational Linguistics (2011)
24. Gupta, M., Zhao, P., Han, J.: Evaluating event credibility on twitter. In: Proceedings of the 2012 SIAM International Conference on Data Mining, pp. 153–164. SIAM (2012)
25. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th International Conference on World Wide Web, pp. 675–684. ACM (2011)
26. Carchiolo, V., Longheu, A., Malgeri, M., Mangioni, G., Previti, M.: Terrorism and war: twitter cascade analysis. In: Del Ser, J., Osaba, E., Bilbao, M.N., Sanchez-Medina, J.J., Vecchio, M., Yang, X.-S. (eds.) IDC 2018. SCI, vol. 798, pp. 309–318. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99626-4_27

27. De Domenico, M., Lima, A., Mougel, P., Musolesi, M.: The anatomy of a scientific rumor. Sci. Rep. **3**, 2980 (2013)
28. Introduction to the tsfeatures package. https://cran.r-project.org/web/packages/tsfeatures/vignettes/tsfeatures.html. Accessed 11 Nov 2019
29. Hyndman, R.J., Wang, E., Laptev, N.: Large-scale unusual time series detection. In: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), pp. 1616–1619. IEEE (2015)
30. Fulcher, B.D., Jones, N.S.: Highly comparative feature-based time-series classification. IEEE Trans. Knowl. Data Eng. **26**(12), 3026–3037 (2014)
31. Nembrini, S., König, I.R., Wright, M.N.: The revival of the Gini importance? Bioinformatics **34**(21), 3711–3718 (2018)
32. Friedman, J.H.: A variable span scatterplot smoother (1984). http://www.slac.stanford.edu/cgi-wrap/getdoc/slac-pub-3477.pdf
33. Bischl, B., et al.: mlr: machine learning in R. J. Mach. Learn. Res. **17**(170), 1–5 (2016)
34. Ribeiro, M.T., Singh, S., Guestrin, C.: Model-agnostic interpretability of machine learning (2016). arXiv preprint arXiv:1606.05386
35. Puri, N., Gupta, P., Agarwal, P., Verma, S., Krishnamurthy, B.: Magix: model agnostic globally interpretable explanations (2017). arXiv preprint arXiv:1706.07160
36. Friedman, J.H., Popescu, B.E., et al.: Predictive learning via rule ensembles. Ann. Appl. Stat. **2**(3), 916–954 (2008)
37. Apley, D.W.: Visualizing the effects of predictor variables in black box supervised learning models (2016). arXiv preprint arXiv:1612.08468
38. Accumulated local effects plot. https://christophm.github.io/interpretable-ml-book/ale.html. Accessed 11 Nov 2019