

Detection and Visualization of Misleading Content on Twitter

Christina Boididou · Symeon Papadopoulos · Markos Zampoglou ·
Lazaros Apostolidis · Olga Papadopoulou · Yiannis Kompatsiaris

Received: date / Accepted: date

Abstract The problems of online misinformation and fake news have gained increasing prominence in an age where user-generated content and social media platforms are key forces in the shaping and diffusion of news stories. Unreliable information and misleading content are often posted and widely disseminated through popular social media platforms such as Twitter and Facebook. As a result, journalists and editors are in need of new tools that can help them speed up the verification process for content that is sourced from social media. Motivated by this need, in this paper we present a system that supports the automatic classification of multimedia Twitter posts into credible or misleading. The system leverages credibility-oriented features extracted from the tweet and the user who published it, and trains a two-step classification model based on a novel semi-supervised learning scheme. The latter uses the agreement between two independent pre-trained models on new posts as guiding signals for retraining the classification model. We analyze a large labeled dataset of tweets that shared debunked fake and confirmed real images and videos, and show that integrating the newly proposed features, and making use of bagging in the initial classifiers and of the semi-supervised learning scheme, significantly improves classification accuracy. Moreover,

This work has been supported by the REVEAL and InVID projects, under contract nr. 610928 and 687786 respectively, funded by the European Commission.

C. Boididou
Urban Big Data Centre Glasgow, UK
E-mail: christina.mpond@gmail.com

C. Boididou, S. Papadopoulos, M. Zampoglou, L. Apostolidis,
O. Papadopoulou, Y. Kompatsiaris
CERTH-ITI, Thessaloniki, Greece
E-mail: {papadop, markzampoglou, laaposto, olgapapa,
ikom}@iti.gr

we present a web-based application for visualizing and communicating the classification results to end users.

Keywords social media · verification · fake detection · information credibility

1 Introduction

Popular social media platforms such as Twitter and Facebook are nowadays an integral part of the journalistic and news diffusion process. This is not only due to the fact that these platforms have lowered the barrier for citizens to contribute to news generation and documentation with their own content, but also due to the possibilities they offer for rapidly disseminating news to one's network of contacts and to broader communities. These new capabilities with respect to publishing and sharing content have led to the uncontrolled propagation of large volumes of news content over social networks. It is now possible for a news story published by an individual to reach huge numbers of readers in very short time. This is especially true for cases where multimedia content (images, videos) is involved. Those often undergo faster and wider sharing (and sometimes become viral) due to the fact that multimedia is easy to consume and is often used as evidence for a story.

The high volume and dissemination speed of news-relevant social media content creates big challenges for the journalistic process of verification. On the one hand, news organizations are constantly looking for original user-generated content to enrich their news stories. On the other hand, having very little time at their disposal to check the veracity of such content, they risk publishing content that is misleading or utterly fake, which would be detrimental to their credibility. For instance, in the case of a breaking story (e.g., natural disaster,

terrorist attack), there is a massive influx of reports and claims, many of which originate from social media. It is exactly this setting where the risk of falsely accepting misleading content as credible is the highest.

As misleading (or, for the sake of brevity, *fake*), we consider any post that shares multimedia content that does not faithfully represent the event that it refers to. This could, for instance, include a) content from a past event that is reposted as being captured in the context of a currently unfolding similar event, b) content that is deliberately manipulated (also known as tampering, doctored or photoshopping), or c) multimedia content that is published together with a false claim about the depicted event. Fig. 1 illustrates a “famous” example of a fake photo that is often recycled after major hurricanes and supposedly depicts a shark swimming in a flooded freeway. It is noteworthy that despite this being a well-known case, there are numerous people who still fall for it (as attested by the number of retweets in each case). In contrast, as *real*, we define posts that share content that faithfully represents the event in question, and can therefore be used in the context of news reporting. There are also in-between cases, such as for instance, posts that debunk fake content or refer to it with a sense of humour. Since those posts are quite obvious for human investigators, but rather hard for automatic classification systems, we consider them to be out of the scope of this work.

The impact of fake content being widely disseminated can be severe. For example, after the Malaysia Airlines flight MH370 disappeared on March 2014, numerous fake images that became viral on social media raised false alarms that the plane was detected¹. This deeply affected and caused emotional distress to people directly involved in the incident, such as the passengers’ families. In another case, on April 2013, a fake tweet was posted by the Associated Press account, which had been hacked for that purpose, stating that the White House had been hit by two explosions and that Barack Obama was injured². This caused the S&P 500 index to decline by 0.9%, which was enough to wipe out \$ 130 billion in stock value in a matter of seconds.

Examples such as the above point to the need for methods that can identify misleading social media content. One of the first such attempts [9] used a supervised learning approach, in which a set of news-related tweets were annotated with respect to their credibility and then used to train a model to distinguish between the two classes; experiments were conducted on a dataset collected around trending news stories and an-



Fig. 1: Examples of fake shark image that was posted several times after major hurricanes in the US (depicted posts refer to Sandy, Matthew, Harvy and Irma).

notated with the help of crowd workers, leading to an accuracy of approximately 86%. However, this level of performance was achieved by performing feature selection on the whole dataset (i.e. both training and test) and by a cross-validation approach that did not ensure full independence between the events included in the training and test sets respectively. Furthermore, some of the employed credibility features, such as the retweet tree of a tweet, are hardly applicable in a real-time setting. Follow-up research on the problem [10] suffered from similar issues, i.e. the “leaking” of information from the training set into the test set, thus giving an optimistic sense of the achievable classification accuracy.

In this paper, which offers an extended presentation and more thorough treatment of our previous work [7], we present an approach that moves beyond the supervised learning paradigm for classifying social media content into credible (*real*) or misleading (*fake*). The proposed approach uses a variety of content-based and contextual features for the social media post in question and builds two classification models that are used to produce two independent first-level predictions regarding the credibility of the post. At a second step, a top-level classifier leverages these first-level predictions on “unseen” content for retraining the best of the first-level models, following a semi-supervised learning paradigm. In that way, the resulting model is well tuned

¹ snopes.com/photos/airplane/malaysia.asp

² www.theguardian.com/business/2013/apr/23/ap-tweet-hack-wall-street-freefall

to the special characteristics of the unseen content and produces more confident predictions. Experiments on a public annotated corpus of multimedia tweets demonstrate the effectiveness of the proposed approach. Additionally, we propose a web-based user interface for visualizing and communicating the result of automatic analysis to end users.

The contributions of this work include the following: 1) the use of a feature set for the representation of users and tweets, extending the ones used by previous studies [1, 9, 10]; 2) the application of an agreement-based retraining, previously proposed in [36] for the task of polarity classification, which allows the model to adapt to new, unknown datasets; 3) an extensive experimental study on a large annotated corpus of tweets investigating the impact of the proposed novelties and comparing with state-of-the-art methods; 4) a web-based application that allows users to test our approach for verification, and to further investigate the role of different features on the verification result.

2 Related Work

The presented work focuses on the problem of *misleading social media content detection*, and more specifically on Twitter posts (tweets) that are accompanied by multimedia content. More precisely, given a single tweet that claims to provide information on an event and contains an image or video to support the claim, our task is to return an estimate of its credibility. Furthermore, given that news professionals are generally reluctant to trust “black box” systems, a second objective is to be able to communicate the system’s output by illustrating which features matter most towards the final estimate. Finally, for the system to be applicable in the real world, it is important to ensure generalization across different events, i.e. to make sure that the system can adapt to new content.

Given the above definition, the examined problem is related but distinct to several other problems. *Hoax detection* [17] is the problem of debunking entire stories posted on the Web. Thus, it deals with larger amounts of text than a single social media post, and it is typically not backed by multimedia evidence. A similar problem is *rumour detection*. A rumour is an unverified piece of information at the time of its publication. Typically, rumours do not directly correspond to a single piece of text or a social media post, but rather to a collection of items that disseminate it. Zubiaga et al. [43] present a survey of approaches for rumor detection, including veracity classification and the collection and annotation of rumor-focused datasets from social media. Finally, a

related problem is *automated fact checking*, which pertains to the classification of sentences into non-factual, unimportant factual and check-worthy factual statements [12]. Fact checking methods rely on structured knowledge from databases, such as FreeBase and DBpedia, which contain entities, events and their relations.

The above problems are distinct from the one examined in this paper. For instance, hoax detection and fact-checking typically operate on different types of inputs than social media posts and commonly concern claims that can be verified via a combination of database cross-checking and reasoning. On the other hand, rumour detection operates on social media content, but considers collections of posts. In contrast, the focus in this paper is on the problem of verifying individual social media posts, typically posted in the context of an unfolding newsworthy event. This is an important differentiating factor, especially in the context of the first moments after a claim (expressed by an individual post) circulates in social media, when there is little or no contextual information available (e.g. comments responding to the claim, networks of retweets).

The particular problem studied in this paper was the focus of the “Verifying Multimedia Use” benchmarking task, which was organized in the context of MediaEval 2015 [2] and 2016 [4]. According to the official task definition, “*given a tweet and the accompanying multimedia item (image or video) from an event that has the profile to be of interest in the international news, return a binary decision representing verification of whether the multimedia item reflects the reality of the event in the way purported by the tweet*”. In a comparative study that we recently conducted [6], we present a detailed comparison among three high-performing approaches on the problem, among which is the approach presented here.

The typical methodology for detecting a misleading social media post is to extract a number of features from it, and classify it using a machine learning algorithm. Typical features can be *text-based*, such as linguistic patterns or the presence of capital letters and punctuation, *user-based*, i.e. information extracted from the profile of the user account who made the post such as age or number of followers/friends, or *interaction-based*, such as the number of responses to the post.

As mentioned in the introduction, the work by Castillo et al. [9] is one of the earliest attempts on the problem. The approach attempted to assess credibility at the event/topic level, i.e. produce a credibility score for an entire set of tweets discussing one event. The extracted features included text-based (e.g. tweet length, fraction of capital letters), user-based (e.g. account age, number of followers), topic-based (number of tweets, num-

ber of hashtags in the topic), and propagation-based, i.e. features describing a tree created from the retweets of a message. Besides the critique that the training and test cases were not entirely independent during the training/cross-validation process, the fact that the approach operates on the event level instead of the tweet level means it is not flexible enough for our task. However, many of the features are directly applicable to our task as well. Similarly, Vosoughi et al. [38] use text-, user-, and propagation-based features for rumour verification on Twitter.

In a work that is directly comparable to the one presented here, Gupta et al. [10] train a system on a set of features in order to classify between tweets sharing fake images and tweets sharing real images on a dataset of tweets from Hurricane Sandy. In that way, tweet classification is used as a first step towards verifying the associated images. However, as mentioned in the introduction, the separation between training and test cases was not adequate for reliably assessing the generalization ability of the method. In a similar work, O’Donovan et al. [22] performed an analysis of the distribution of various features within different contexts to assess their potential use for credibility estimation. However, their analysis remains preliminary in the sense that they only analyze feature distributions and not their effectiveness on the classification task. In our work, we move one step further by directly analyzing the performance of different configurations and variations of our approach. More recently, Wu et al. [39] presented a classifier trained on posts from the Chinese micro-blogging platform Sina Weibo. Besides typical features, the paper presents a “propagation tree” that models the activity following a post (reposts, replies). This, however, is only applicable long time after a post is published, once a sufficiently large propagation tree is formed.

Another recent approach is that of Volkova et al. [37], where Twitter posts are classified into “suspicious” vs “trusted” using word embeddings and a set of linguistic features. However, the separation between the two classes is made based on the source, i.e. by contrasting a number of trusted accounts to various biased, satirical, or propaganda accounts. This approach likely ends up classifying the writing styles of the two distinct types of account, while in our case no distinction between trusted and non-trusted accounts was made during model building. Similarly, Rubin et al. [29] use satirical cues to detect fakes, which only applies to a specific subset of cases. Another category of methods attempt to include image features in the classification, under the assumption that the image accompanying a post may carry distinct visual characteristics that differ between fake and real posts [14, 34]. While this assump-

tion may hold true when contrasting verified posts by news agencies to fake posts by unverified sources, it certainly cannot assist us when comparing user-generated fake and real posts. One typical example is fake posts that falsely share a real image from a past event and claim that it was taken from a current one. In this case, the image itself is real and may even originate from a news site, but the post as a whole is fake.

Since we are dealing with multimedia tweets, one seemingly reasonable approach would be to directly analyze the image or video for traces of digital manipulation. To this end, the field of multimedia forensics has produced a large number of methods for tampering detection in images [23, 31, 42] and videos [24] in the recent years. These include looking for (often invisible) patterns or discontinuities that result from operations such as splicing [42], detecting self-similarities that suggest copy-move/cloning attacks [31], or using near-duplicate search to build a history of the various alterations that an image may have undergone in its past (“image phylogenies”) [23]. However, such methods are not well-suited for Web and social media images, for a number of reasons:

- Splicing detection algorithms are often not effective with social media images, as these typically undergo numerous transformations (resaves, crops, rescales), which eliminate the tampering traces.
- Building an image phylogeny requires automatically crawling the Web for all instances of an image, which is an extremely costly task.
- It is highly likely that an image may convey false information without being tampered. Such is the case, e.g. of posting an image from a past event as breaking news, or of misrepresenting the context of an authentic image.

Therefore, an image disseminating false information in social media may no longer contain any detectable traces of tampering, or it may even be untampered in the first place. For that reason, we turn to the analysis of tweet- and user-based features for verification.

Finally, an important aspect of the problem is not only to be able to correctly classify tweets, but also to present verification results to end users in a manner which is understandable and can be trusted by end users. Currently, there exist a few online services aiming to assist professionals and citizens with verification. The Truthy system [26] is a Web service that tracks political memes and misinformation on Twitter, aiming to detect political astroturfing, i.e. organized posting of propaganda disguised as grassroots user contributions. Truthy collects tweets, detects emerging memes, and provides annotation on their truthfulness based

on user manual annotation. RumorLens [28] is a semi-automatic platform combining human effort with computation to detect new rumours in Twitter. Twitter-Trails [20] tracks rumour propagation on Twitter. There also exist some fully automatic tools, such as Tweet-Cred [11] which returns credibility scores for a set of tweets, and Hoaxy [30], a platform for detecting and analysing online misinformation. Finally, with respect to analyzing multimedia content, there are two notable tools: a) the REVEAL Image Verification Assistant [41], which exposes a number of state-of-the-art image splicing detection algorithms via a web-user interface, and b) the Video News Debunker [35], which was released by the InVID project as a Chrome plugin, to assist investigators in verifying user-generated news videos.

3 Misleading Social Media Content Detection

Fig. 2 depicts the main components of the proposed framework. It relies on two independent classification models built on the training data using two different sets of features, tweet-based (TB) and user-based (UB). Model bagging is used to produce more reliable predictions based on classifiers from each feature set. At prediction time, an agreement-based retraining strategy is employed (*fusion*), which combines the outputs of the two bags of models in a semi-supervised learning manner. The verification result is then visualized to end users. The training of classification models and a set of feature distributions that are used by the visualization component are based on an annotated set of tweets, the so-called *Verification Corpus*, which is further described in Section 4. The implementation of the framework and the corpus are publicly available on GitHub^{3,4}.

3.1 Feature extraction and processing

The design of features used in our framework was carried out following a study of the way in which news professionals, such as journalists, verify content on the Web. Based on relevant journalistic studies, such as the study of Martin et al. [19], and the Verification Handbook [32], as well as on previous similar approaches [9, 10], we defined a set of features that are important for verification. These are not limited to the content itself, but also pertain to its source (Twitter account that made the post) and to the location where it was posted. We decided to avoid multimedia forensics features following the conclusion of our recent study [40]

that the automatic processing of embedded multimedia on Twitter remove the bulk of forensics-relevant traces from the content. This was also confirmed by our recent MediaEval participations [3, 5], where the use of forensics features did not lead to noticeable improvement. The feature extraction process produces a set of TB and UB features for each tweet, which are presented in Table 1 and described below.

Tweet-based features (TB): We consider four types of feature related to tweets: a) text-based, b) language-specific, c) Twitter-specific, and d) link-based.

a) text-based: These are extracted from the text of the tweet, and include simple characteristics (length of text, number of words), stylistic attributes (number of question and exclamation marks, uppercase characters), and binary features indicating the existence or not of emoticons, special words (“please”) and punctuation (colon).

b) language-specific: These are extracted for a predefined set of languages (English, Spanish, German), which are detected using a language detection library⁵. They include the number of positive and negative sentiment words in the text using publicly available sentiment lexicons: for English we use the list by Jeffrey Breen⁶, for Spanish the adaptation of ANEW [27] and for German the Leipzig Affective Norms [15]. Additional binary features indicate whether the text contains personal pronouns (in the supported languages), and the number of detected slang words. The latter is extracted using lists of slang words in English⁷ and Spanish⁸. For German, no available list was found and hence no such feature is computed. Moreover, the number of nouns in the text was also added as feature, and computed based on the Stanford parser only for English [16]. Finally, we use the *Flesch Reading Ease* method⁹ to compute a readability score in the range [0: hard-to-read, 100: easy-to-read]. For tweets written in languages where the above features cannot be extracted, we consider their values missing.

c) twitter-specific: These are features related to the Twitter platform, including the number of re-tweets, hashtags, mentions, URLs and a binary feature expressing whether any of the URLs points to external (non-Twitter) resources.

d) link-based: These include features that provide information about the links that are shared through the tweet. This set of features is common in both the TB

⁵ code.google.com/p/language-detection/

⁶ github.com/jeffreymbreen/

twitter-sentiment-analysis-tutorial-201107

⁷ onlineslangdictionary.com/word-list/0-a/

⁸ www.languagerealm.com/spanish/spanishslang.php

⁹ github.com/ipeirotis/ReadabilityMetrics

³ github.com/MKLab-ITI/computational-verification

⁴ github.com/MKLab-ITI/image-verification-corpus

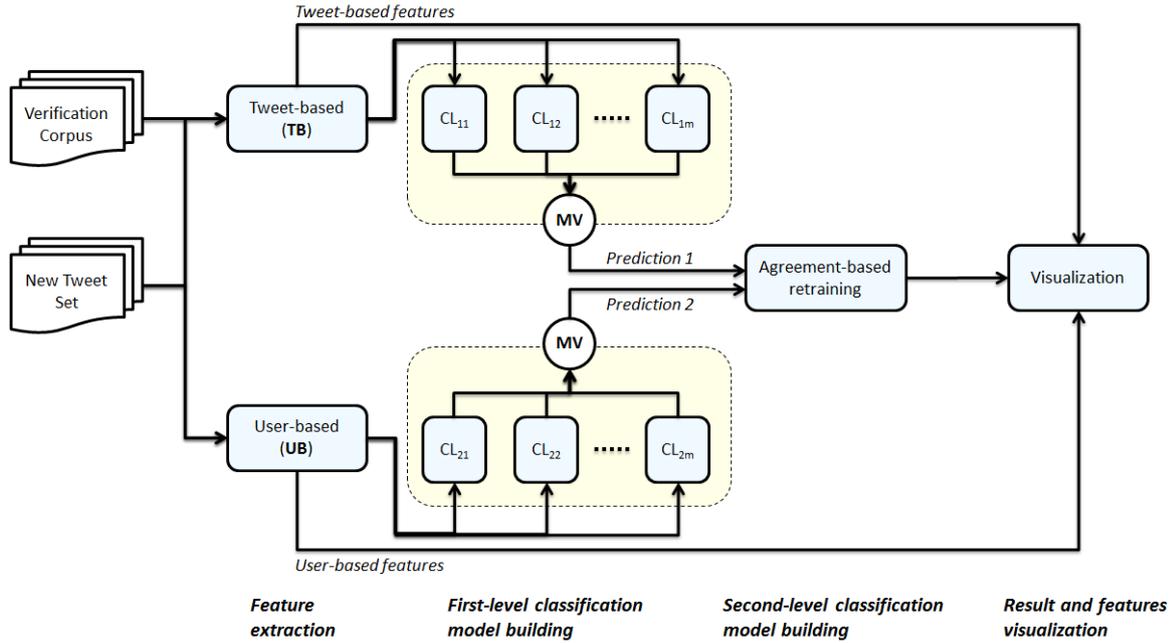


Fig. 2: Overview of the proposed framework. MV stands for Majority Voting.

Table 1: Overview of verification features. Link-based features are extracted in the TB case for external links that tweets may share, and in the UB case for the URL included in the account profile. Features with an asterisk were proposed in [1, 10] and will be referred to as *Baseline Features* (BF), while the full feature set (union of BF and newly proposed ones) will be referred to as *Total Features* (TF).

Tweet-based Features (TB)		User-based Features (UB)	
<i>text-based</i>		<i>user-specific</i>	
#words*	has ‘‘please’’	#friends*	has location
length of text*	has colon	#followers*	has existing location
#question marks*	contains happy emoticon*	follower-friend ratio*	has bio description
#exclamation marks*	contains sad emoticon*	#tweets	tweet ratio
contains question mark*	#uppercase chars*	#media content	account age
contains exclamation mark*		has profile image	is verified*
		has header image	#times listed*
		has a URL*	
<i>language-specific</i>		<i>link-based (common for TB and UB)</i>	
#pos senti words*	contains 1st pers.pron.*	WOT score	alexa country rank
#neg senti words*	contains 2nd pers.pron.*	in-degree centrality	alexa delta rank
#slangs	contains 3rd pers.pron.*	harmonic centrality	alexa popularity
#nouns	readability		alexa reach rank
<i>twitter-specific</i>			
#retweets*	#mentions*		
#hashtags*	#URLs*		
has external link			

and UB sets, but in the latter it is defined in a different way (see link-based category in UB features). For TB, depending on the existence of an external URL in the tweet, its reliability is quantified based on a set of Web metrics: i) the WOT score¹⁰, which is a way to assess the trust on a website using crowdsourced reputation

ratings, ii) the in-degree and harmonic centralities¹¹, computed based on the links of the Web graph, and iii) four Alexa metrics (rank, popularity, delta rank and reach rank) based on the rankings API¹².

¹⁰ www.mywot.com/

¹¹ wwwranking.webdatacommons.org/more.html

¹² data.alexa.com/data?cli=10&url=google.com

User-based features (UB): These are related to the Twitter account posting the tweet. We divide them into a) user-specific and b) link-based features.

a) user-specific: These include the user’s number of friends and followers, the account age, the follower-friend ratio, the number of tweets by the user, the tweet ratio (number of tweets/day divided by account age) and several binary features: whether the user is verified by Twitter, whether there is a biography in his/her profile, whether the user declares his/her location using a free text field, and whether the location text can be parsed into an actual location¹³, whether the user has a header or profile image, and whether a link is included in the profile.

b) link-based: In this case, depending on the existence of a URL in the Twitter profile description, we apply the same Web metrics as the ones used in the link-based TB features. If there is no link in the profile, the values of these features are considered to be missing.

After feature extraction, the next steps include pre-processing, cleaning and transformation. To handle the issue of missing values on some of the features, we use linear regression for estimating their values: we consider the attribute with the missing value as a dependent (class) variable and apply linear regression for numeric features. The method cannot support the prediction of boolean values and hence those are left missing. Only feature values from the training set are used in this process. Data normalization is also performed to scale the numeric feature values to the range $[-1, 1]$.

3.2 Building the classification models

We use the TB and UB features to build two independent classifiers (CL_1 , CL_2 respectively), each based on the respective set of features. To further increase classification accuracy, we make use of bagging: we create m different subsets of tweets from the training set, including equal number of samples for each class (some samples may appear in multiple subsets), leading to the creation of m instances of CL_1 and CL_2 ($m = 9$ in our experiments). These are denoted as $CL_{11}, CL_{12}, \dots, CL_{1m}$ and $CL_{21}, CL_{22}, \dots, CL_{2m}$ respectively in Fig. 2. The final prediction for each of the test samples is calculated using the average of the m predictions. Concerning the classification algorithm, we tried both Logistic Regression (LR) and Random Forests (RF) of 100 trees.

3.3 Agreement-based retraining

A key contribution of the proposed framework is the introduction of an *agreement-based retraining* step (the fusion block in Fig. 2) as a second-level classification model for improving the generalization ability of the framework to new content. The agreement-based retraining step was motivated by recent work on social media sentiment analysis that was demonstrated to effectively address the problem of out-of-domain polarity classification [36].

In our implementation, we combine the outputs of classifiers CL_1 , CL_2 as follows: for each sample of the test set, we compare their outputs and depending on their agreement, we divide the test set in the *agreed* and *disagreed* subsets. The elements of the *agreed* set are assigned the agreed label (fake/real) assuming that it is correct with high likelihood, and they are then used for retraining the best performing of the two first-level models (CL_1, CL_2)¹⁴ to re-classify the *disagreed* elements. Two retraining techniques are investigated: The first is to use just the agreed samples to train the CL classifier (denoted as CL^{ag}), while the second is to use the entire (total) set of initial training samples extending it with the set of agreed samples (denoted as CL^{tot}). The goal of retraining is to create a new model that is tuned to the specific data characteristics of the new content. The resulting model is expected to predict more accurately the values of the samples for which CL_1, CL_2 did not initially agree. In the experimental section, we test both of the above retraining variants.

3.4 Verification result visualization

The main idea behind the visualization of the produced verification output is to present it along with the list of credibility features that were extracted from the input tweet and the user account that posted it, and to give to end users the option to select any of these features and inspect its value in relation to the distribution that this feature has for real versus fake tweets, as computed with respect to the verification corpus (Section 4).

Fig. 3 depicts an annotated screenshot of this application, which is publicly available¹⁵. In terms of usage, the investigator first provides the URL or id of a tweet of interest, and then the application presents the extracted tweet- and user-based features and the verification result (fake/real) for the tweet in the form of a color-coded frame (red/green respectively) and a bar.

¹⁴ The selection is based on their performance on the training set during cross-validation.

¹⁵ reveal-mklab.iti.gr/reveal/fake/

¹³ Using: github.com/socialsensor/geo-util

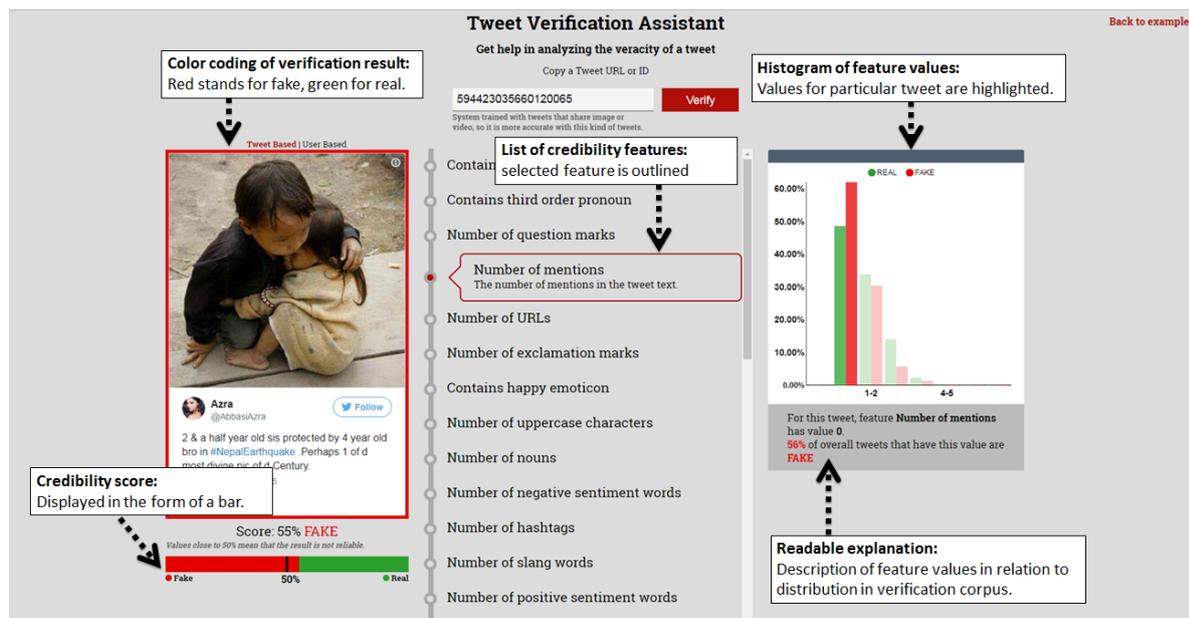


Fig. 3: Snapshot of the Tweet Verification Assistant interface. Given a tweet, a user can explore the verification result, including the extracted feature values and their distribution on the Verification Corpus.

It also offers the possibility of inspecting the feature values in the central column. By selecting a feature, its value distribution appears at the right column, separately for fake and real tweets (side-by-side). Moreover, a textual description informs the user about the percentage of tweets of this class (fake or real) that have the same value for this feature. In that way, the investigator may better understand how the verification result is justified based on the individual values of the features in relation to the “typical” values that these features have for fake versus real tweets.

4 Verification Corpus

Our fake detection models are based on a publicly available verification corpus (VC) of fake and real tweets that we initially collected for the needs of organizing the MediaEval 2015 Verifying Multimedia Use (VMU) task [2]¹⁶. This consists of tweets related to 17 events (or hoaxes) that comprise in total 193 cases of real images, 218 cases of misused (fake) images and two cases of misused videos, and are associated with 6,225 real and 9,404 fake tweets posted by 5,895 and 9,025 unique users respectively. The list of events and some basic

¹⁶ The VC was since then expanded with new data that was used as part of the VMU 2016 task. However, the bulk of the experiments reported here refer to the 2015 version of the data, so any reference to VC refers to the 2015 edition of the dataset, unless otherwise stated.

statistics of the collection are presented in Table 2. Several of the events, e.g., Columbian Chemicals, Passport Hoax and Rock Elephant, were actually hoaxes, hence all content associated with them is fake. Also, for several real events (e.g., MA flight 370) no real images (and hence no real tweets) were included in the dataset, since none came up as a result of the data collection. Fig. 4 illustrates four example cases that are characteristic of the types of fake in the corpus. These include reposting of past images in the context of a new event, computer generated imagery, images accompanied by false claims, and digitally tampered images.

The set of tweets T of the corpus was collected with the help of a set of keywords K per event. The ground truth labels (fake/real) of these tweets were based on a set of online articles that reported on the particular images and videos. Only articles from reputable news providers were used that adequately justified their decision about the veracity of each multimedia item. This led to a set of fake and real multimedia cases, denoted as I_F , I_R respectively, where each multimedia case is represented by a URL pointing to an instance of the considered multimedia content. These were then used as seeds to create the reference verification corpus $T_C \subset T$, which was formed by tweets that contain at least one item (URL) from the two sets. In order not to restrict the corpus to only those tweets that point to the exact seed URLs, a visual near-duplicate search technique was employed [33] to identify tweets that contained images



Fig. 4: Types of fake: (i) reposting of real photo depicting two Vietnamese siblings as being captured during the Nepal 2015 earthquakes; (ii) reposting of artwork as a photo from Solar Eclipse of March 2015; (iii) speculation of someone as being suspect of the Boston Marathon bombings in 2013; (iv) spliced sharks on a photo captured during Hurricane Sandy in 2012.

that were found to be highly similar with any item in the I_F or the I_R set. To ensure near-duplicity, a minimum threshold of similarity was empirically set, tuned for high precision. A small amount of the images exceeding the threshold were manually found to be irrelevant to the ones in the seed set and were then removed.

The corpus was further cleaned in two ways: a) we considered only unique posts by eliminating re-tweets, since their tweet-based features would be identical; b) by manual inspection, we ensured that no posts were included that featured humorous content, nor posts that declared that their content is fake, both of which cases would be hard to classify as either real or fake.

As the aim of our work is to assess the generalization capability of the fake detection framework, we used every tweet in the corpus regardless of language. The aim has been to use a comprehensive corpus, which contains the widest possible variety of fake tweets even though this complicates the machine learning process due to missing feature values as explained in Section 3.1.

5 Experimental Study

5.1 Overview

The aim of the conducted experiments was to evaluate the classification accuracy of different models on samples from new (unseen) events. We consider this an important aspect of a verification framework, as the nature of untrustworthy (fake) tweets may vary across different events. Accuracy is computed as the ratio of correctly classified samples (N_c) over total number of test samples (N): $a = N_c/N$. The initial design of the evaluation scheme was thought of as a kind of *event-based cross-validation*: for each event E_i of the 17 events

in the VC , we intended to use the remaining 16 events for training and E_i for testing. Each of these 17 potential splits is denoted as T_i . However, as shown in Table 2, many events only contain fake tweets, while others have very few tweets in total. These are unsuitable for evaluations, thus we chose to focus on events E1, E2, E12, and E13 for the results presented here. We also consider an additional split, which was proposed by the MediaEval 2015 VMU task [2], in which events $E1-E11$ are used for training, and events $E12-E17$ are used for testing. This makes it possible to compare our performance with the one that was achieved by methods that participated in the task. Finally, another test run is the one used in MediaEval VMU 2016, in which all 17 events are used for training, and a new, independent set of tweets used for evaluation. The latter two splits are denoted as VMU 2015 and 2016 respectively.

5.2 New Features and Bagging

We first assess the contribution of the new features and bagging to the accuracy of the framework. To this end, we build the CL_1 , CL_2 classifiers with and without the bagging technique. To create the models without bagging, we selected each time an equal number of random fake and real samples for training. We applied this procedure both for the Baseline (BF) and Total Features (TF) (cf. Table 1 caption). Table 3 presents the average accuracy for each setting.

We observe that the use of bagging led to considerably improved accuracy for both CL_1 and CL_2 . In addition, further improvements are achieved when using the TF features over BF. We see that bagging led to an absolute improvement of approximately 10% and 4% in the accuracy of CL_1 and CL_2 respectively (when using

Table 2: List of events in *VC-MediaEval 2015*: For each event, we report the number of unique real (if available) and fake cases of multimedia (I_R , I_F respectively), unique tweets that shared those media items (T_R , T_F) and Twitter accounts that posted the tweets (U_R , U_F).

ID	Name	I_R	T_R	U_R	I_F	T_F	U_F
E1	Hurricane Sandy	148	4,664	4,446	62	5,558	5,432
E2	Boston Marathon bombing	28	344	310	35	189	187
E3	Sochi Olympics	-	-	-	26	274	252
E4	Bring Back Our Girls	-	-	-	7	131	126
E5	MA flight 370	-	-	-	29	310	302
E6	Columbian Chemicals	-	-	-	15	185	87
E7	Passport hoax	-	-	-	2	44	44
E8	Rock Elephant	-	-	-	1	13	13
E9	Underwater bedroom	-	-	-	3	113	112
E10	Livr mobile app	-	-	-	4	9	9
E11	Pig fish	-	-	-	1	14	14
E12	Nepal earthquake	11	1,004	934	21	356	343
E13	Solar Eclipse	4	140	133	6	137	135
E14	Garissa Attack	2	73	72	2	6	6
E15	Samurai and Girl	-	-	-	4	218	212
E16	Syrian Boy	-	-	-	1	1,786	1,692
E17	Varoufakis and ZDF	-	-	-	1	61	59
	Total	193	6,225	5,895	220	9,404	9,025

the TF features), while the use of TF features over BF to an improvement of approximately 22% when bagging is used. Combined, the use of bagging and the newly proposed features led to an absolute improvement of approximately 25% and 30% for CL_1 and CL_2 respectively. Given the clear benefits of using bagging, in subsequent experiments, all reported results refer to classifiers with bagging and TF.

Table 3: Performance of CL_1 , CL_2 , and effect of bagging and Total Features (TF) over Baseline Features (BF).

	CL_1	CL_1 -bag	CL_2	CL_2 -bag
BF	64.04	67.63	51.94	58.00
TF	79.64	89.53	77.59	81.67

5.3 Agreement-based retraining technique

We use the entire set of features (TF) for assessing the accuracy of the agreement-based retraining approach. Table 4 shows the scores obtained separately for various splits. In the table, we do not present events that only contain fake tweets, as well as those with too few tweets, and as a result, only results for splits T1, T2, T12, and T13 are presented. Additionally, we present the average accuracy for these four events, as well as the average across all 17 events. We also present the accuracy obtained on the VMU 2015 and 2016 splits. All results in Table 4 are given for both the Logistic Regression (LR) and the Random Forest (RF) classifiers. The first two

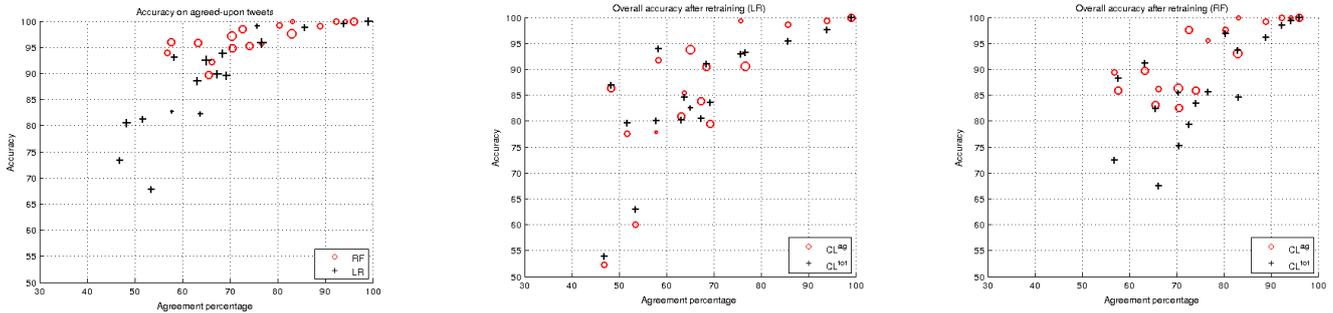
columns present the results using only CL_1 (i.e. Tweet-Based), while the next two present results from CL_2 (i.e. User-Based). These are similar in concept to previously tested supervised learning approaches [1, 10]. The following two columns present the accuracy achieved using simple concatenation of the user-based and tweet-based feature vectors into a single-level classifier (CL^{cat}). The last four columns give the overall accuracy for the two agreement-based retraining models (CL^{ag} and CL^{tot}).

Comparing the scores of the CL_1 and CL_2 classifiers with those of the agreement-based retraining variations, one can see in most cases a clear improvement in terms of classification accuracy (more than 5% on average across all events). Another observation is that, while on average simple concatenation performs worse than agreement-based retraining, it outperforms agreement-based classifiers on VMU 2015. Furthermore, it performs marginally worse on VMU 2016 compared to CL^{ag} using LR, our best-performing method on that split. However, on average the agreement-based methods perform significantly better on most splits and on average, demonstrating a greater robustness compared to fusion using simple concatenation. With respect to the comparison between Logistic Regression and Random Forests, while the results are comparable in many cases, LR performs better overall.

To further analyze the behaviour of the classification retraining approach, we study the relation between the percentage of tweets where the two classifiers agree and the respective classification accuracy. From the three scatter plots of Fig. 5, the first shows the accuracy within the agreed set in relation to the percentage of

Table 4: Accuracy for the entire set of features TF . Agreement levels between the CL_1 , CL_2 and the CL^{cat} classifiers (all with the use of bagging) and overall accuracy for each model (CL^{ag} , CL^{tot}) and each split.

	CL1		CL2		CL ^{cat}		CL ^{ag}		CL ^{tot}	
	LR	RF	LR	RF	LR	RF	LR	RF	LR	RF
T1	71.49	72.57	83.88	93.74	78.86	73.25	93.84	86.27	82.49	85.43
T2	74.63	67.84	79.19	90.36	82.89	71.07	83.81	89.70	80.53	91.14
T12	85.46	80.83	43.99	71.46	86.01	57.81	86.38	83.10	86.91	82.35
T13	78.84	65.85	69.82	87.08	82.02	63.36	80.97	85.88	80.14	88.27
Average	77.60	71.77	69.22	85.66	82.45	66.37	86.25	86.24	82.52	86.80
Average T1-T17	82.82	87.28	71.68	84.36	86.86	80.69	85.14	92.46	84.62	87.04
VMU 2015	88.97	89.53	72.35	81.67	92.36	78.88	90.57	90.84	90.58	90.57
VMU 2016	91.75	87.42	71.23	88.81	93.27	83.92	93.87	90.73	92.72	90.64


 Fig. 5: Scatter plots of percentage of agreed tweets and classification accuracy for all splits T_i . Left: Accuracy for agreed tweets for LR and RF. Center: Overall accuracy following retraining using LR. Right: Overall accuracy following retraining using RF. Marker sizes are proportional to number of items in the respective training set.

agreed tweets in the test set. While a correlation can be seen for both RF and LR, the former seems to perform better overall, revealing a greater consistency between CL_1 , and CL_2 . In contrast, results for LR are more scattered, showing both lower agreement rates for many events, as well as reduced accuracy within the agreed set for these cases. The next two plots show the final accuracy following retraining, using both methods. In this case, while LR (centre) seems to demonstrate a greater spread between events, on average it performs better than RF (right). Thus, while LR is not as consistent in terms of agreement between CL_1 and CL_2 , it more than makes up for it in the retraining step. With respect to the two retraining approaches, LR with CL^{tot} performs in many cases better than CL^{ag} . The opposite is true for RF. In this case, CL^{ag} seems to perform better in many cases. In combination with the results of Table 4 this implies that, to an extent, the performance of the retraining approach is partly dependent on the underlying classification algorithm.

5.4 Performance on different languages

We also assessed the classification accuracy of the framework for tweets written in different languages, i.e. the extent to which the framework is language-dependent.

We considered the five most used languages in the corpus (by number of tweets). Note that in many cases no language is detected, either because the text contains no text but just hashtags/URLs or the length of the text is too small for the language detector. For this reason, we consider a category of tweets denoted as **NO-LANG**, and compare between the following cases: English (**EN**), Spanish (**ES**), no language (**NO-LANG**), Dutch (**NL**) and French (**FR**). Table 5 shows the languages tested and the corresponding number of samples.

Using the total amount of features (TF), we computed the accuracy for the VMU 2015 and 2016 sets, separately for each language. Table 6 shows the results. It can be seen that results exhibit greater variance on VMU 2015 in terms of accuracy across languages: in French, the accuracy of the RF classifier is very low (63-65%), while LR does not seem to suffer from a similar problem (88-92%). The results for VMU 2016 are more consistent across language, even though French still exhibits lower performance. Besides French,

Table 5: Number of tweets for most frequent languages on *VC* (including the set of tweets where no language could be detected (NO LANG)).

	VMU 2015		VMU 2016	
	#training	#testing	#training	#testing
EN	8,713	3,516	12,229	1,558
ES	1,017	86	1,103	146
FR	231	33	264	181
NO-LANG	692	37	729	51
NL	239	23	262	63

Table 6: Language-based accuracy based on the TF features and the two retraining variations, CL^{ag} and CL^{tot} .

	VMU 2015			
	CL^{ag}		CL^{tot}	
	LR	RF	LR	RF
EN	90.53	91.00	90.53	90.43
ES	96.16	86.40	97.67	89.07
FR	88.79	65.15	92.73	63.64
NL	96.52	96.96	94.35	96.52
NO-LANG	88.92	81.08	88.92	83.78
	VMU 2016			
EN	93.80	83.94	93.07	89.24
ES	98.01	88.63	97.79	91.78
FR	88.95	83.20	90.72	81.66
NL	93.49	89.52	96.83	88.73
NO-LANG	88.43	90.98	96.27	93.33

the other cases for which we do not extract language-specific features (NL and NO-LANG) do not seem to suffer from reduced performance. This is encouraging since it indicates that the framework can in many cases work even with languages for which the language-specific features are not defined. However, the low performance on French, contrasted with the high success rates on Dutch for which the number of examples is nearly equal, implies that there may be language-specific nuances that should be further explored in our future work.

5.5 Comparison with state of the art methods

We also compare our method with the ones submitted to the 2015 and 2016 editions of the MediaEval VMU task. For 2015, these include the systems by UoS-ITI [21], MCG-ICT [13], and CERTH-UNITN [3]. For 2016, these include IRISA-CNRS [18], MCG-ICT [8], UNITN [25], and a run inspired by the TweedCred algorithm [11] that we implemented on our own. For each of the competing MediaEval approaches, we compare against their best *run*¹⁷. The comparison is done using the F1-score, which is the official metric of the task. Note that for 2016, we trained our approach using the provided development set (which was used by all competing systems

in the same year). This consists of the whole MediaEval VMU 2015 dataset (Table 2).

According to the results, the best variant of the proposed method achieves the second best performance ($F = 0.935$) on the VMU 2015 task, reaching almost equal performance to the best run by MCG-ICT [13] ($F = 0.942$). In the more challenging VMU 2016 task, the MCG-ICT approach performs considerably worse, while the proposed method retains its high performance, and achieves the best performance achieving an F-score of 0.944, better than the best approach at the time, by IRISA-CNRS ($F = 0.924$) [18].

In both years, our method performs similarly to the best method. However, we should take into account how these respective methods operate and why -in contrast to our approach- they may not be able to perform equally well in many real-world situations. The characteristic of the task that these algorithms leverage to reach their high performance is that, for each event, the dataset contains multiple tweets sharing the same multimedia item. Even more so, the VC also groups similar (non-identical) images together using near-duplicate search. Both algorithms take advantage of this information by first finding all tweets that share the same item, and then using the aggregated set for classification on a cluster basis. Specifically, MCG-ICT relies on a model that first clusters tweets into topics according to the multimedia resource that they contain and then extracts topic-level features for building the fake detection classifier. Similarly, the IRISA-CNRS method aggregates all tweets sharing the same image, and then produces an estimate for all of them by searching for telltale patterns (e.g., “photographed by”) in the tweets or references to known sources. In the more challenging 2016 setting, MCG-ICT cannot perform as well, but the trustworthy source detection of IRISA-CNRS succeeds, since it commonly ends up finding at least one tweet per image providing the necessary verification clue.

In real-world cases, the above approaches may work well when a fake image has already been circulating for some time, and multiple tweets can be found sharing it - especially if an investigator undertakes the effort of performing reverse image search and aggregating all variants of the image and the tweets that share it. In such cases, the IRISA-CNRS method imitates the behaviour of a human investigator, by searching through all these instances of the same item and detecting the most informative ones for verification. However, especially in the case of breaking news, it is very common to come across a single image that no-one else has shared yet, and that poses as a real photo/clip from an unfolding event. In such cases, it is impossible to apply these methods. Our own approach, in contrast, can operate on a per-tweet

¹⁷ In MediaEval, each team can submit up to five runs.

Table 7: Comparison between the proposed method and the best MediaEval VMU 2015 and 2016 submissions.

VMU 2015			VMU 2016		
Method		F1	Method		F1
UoS-ITI [21]		0.830	IRISA-CNRS [18]		0.924
MCG-ICT [13]		0.942	MCG-ICT [8]		0.683
CERTH-UNITN [3]		0.911	UNITN [25]		0.870
			TweedCred [11]		0.720
CL ^{ag}	LR	0.932	CL ^{ag}	LR	0.944
	RF	0.935		RF	0.909
CL ^{tot}	LR	0.932	CL ^{tot}	LR	0.935
	RF	0.933		RF	0.908

basis with robust performance, and exploits the retraining step as soon as a collection of event-specific tweets (without necessarily sharing the same multimedia content) is available. This makes the method more practical in a wide variety of settings. Thus, we consider the fact that we manage to achieve comparable results to both competing methods by individually classifying tweets to be indicative of the increased robustness and practical value of the proposed approach.

5.6 Qualitative analysis

To better understand the strengths and weaknesses of the proposed approach, we also carried out a qualitative analysis of the results, focusing on those cases where the system failed to correctly classify a post. Due to the agreement-based retraining step, one major distinction can be made between agreed and disagreed tweets. For the first case, in which both feature sets (TB and UB) led to the same conclusion, the number of failures is small, as also attested by the first scatter plot of Fig. 5 (especially for RF classification). From this small number of failures, the first two rows of Table 8 present two examples. The first comes from an account with a very small number of followers, but with a significant number of past tweets, and a clear, convincing language in the post. In contrast, the second row gives an example of a false positive, where a syntactically weak post, with capital letters and a question mark was labeled as fake. The four last rows of Table 8 present examples of failures resulting from the agreement-based retraining process on disagreed samples. It is indicative that for both true and fake cases, the prediction (credibility) scores produced by the approach are significantly more extreme here (high and low respectively). In all examples of Table 8, a high score means that the classifier estimated the tweet to be fake, while a low score corresponds to an estimate that the tweet is real.

5.7 Verification visualization

To demonstrate the utility of the web-based verification application, we present two example case studies. In the first, the proposed visualization approach is used on a tweet that shared fake multimedia content in the context of the March 2016 terrorist attacks in Brussels. The tweet (Fig. 6) claimed that the shared video depicted one of the explosions in Zaventem airport, but the video is actually from another explosion in a different airport a few years ago. In the second, a building in Nepal is correctly reported to have collapsed during the 2015 earthquake.

Indeed, the proposed classification framework flags the tweets as fake and real respectively, and presents the feature distributions in order to offer insights about the reasons for its results. Fig. 6 presents the results, including three sample tweet- and user-based feature distributions for each tweet, in the upper and lower part respectively. In the first example, for the fake tweet the number of hashtags is shown to be zero and at the same time the respective bar is highlighted. The plot informs that 63% of the overall training tweets with this value are fake, a fact that partially justifies the classification result. In the next two plots that display the number of mentions and text length, similar conclusions can be made about the veracity of the tweet. In the user-based feature value distributions, the date of creation, number of friends and followers/friends ratio seem to give additional strong signals regarding the low credibility of the account, and the resulting low credibility of the posted tweet. Similar conclusions with respect to the veracity of the second tweet can be drawn from its corresponding distributions, such as, for example the length of the tweet or the number of tweets posted by its author.

6 Conclusions and future work

We presented a robust and effective framework for the classification of Twitter posts into credible versus misleading. Using a public annotated verification corpus, we provided evidence of the high accuracy that the proposed framework can achieve over a number of events of different magnitude and nature, as well as considerable improvements in accuracy as a result of the newly proposed features, the use of bagging, and the application of an agreement-based retraining method that outperforms standard supervised learning. We also demonstrated the utility of a novel visualization approach for explaining the verification result.

To use the proposed approach in real-time settings, one should be cautious of the following caveat. The agreement-based retraining method requires a number

Table 8: Indicative examples of failures for agreed and disagreed samples. Column “Score” presents the prediction (credibility) score produced by the method, with 0 corresponding to predicting the tweet to be real and 1 to be fake. The top two rows present failed cases of agreed tweets, while the remaining four cases of disagreed tweets.

Label	Score	Text	Followers	Tweets
Fake	0.40	Boko Harams Shekau finally resurfaces, orders members to surrender(Video) https://t.co/BvagEB1pD2 https://t.co/QDkjcgQWno	49	2,609
Real	0.56	@RT.com: MORE: Police say explosion also took place in a bar near #Paris stadium https://t.co/y7P3ZsZ05F https://t.co/JTGQiqn2Wj #ISIS?	2,852	16,260
Real	0.60	RT @MariePobemer: RT @Breaking3zero: Fusillades de #Paris : bilan provisoire de 30 morts (AFP) https://t.co/4Gjyc75r1b	440	28,684
Real	0.70	@itsbm_ @DrAlnefisi @MdUbaida #Madaya https://t.co/cLBUuBthoH	2,354	19,491
Fake	0.15	Gabriele_Corno : Lenticular clouds over Fuji #Japan #fuji #lake #cloud ... http://t.co/Eet1wTc1gh http://t.co/eaJ4Q9vayh	737	23,878
Fake	0.31	RT @RAPIDTWITTER Husband Gave His Unfaithful Ex-Wife Half Of Everything He Owned - Literally http://t.co/Rd9pYGstYE http://t.co/RA3Wr9Guta	192,425	70,854

of samples from the new event in order to be applied effectively. Hence, for the first set of arriving items, it is not possible to rely on this improved step. Yet, the rate at which new items arrive in the context of breaking news events could quickly provide the algorithm with a sufficient set of tweets.

In the future, we are interested in looking further into the real-time aspects of credibility-oriented content classification, and conduct experiments that better simulate the problem as an event evolves. We also plan to conduct user studies to test whether the proposed visualization is understandable and usable by news editors and journalists. Finally, we would also like to extend the framework to be applicable to content posted on platforms other than Twitter.

References

- Boididou C, Papadopoulos S, Kompatsiaris Y, Schifferes S, Newman N (2014) Challenges of computational verification in social multimedia. In: Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion, pp 743–748
- Boididou C, Andreadou K, Papadopoulos S, Dang-Nguyen DT, Boato G, Riegler M, Kompatsiaris Y (2015) Verifying multimedia use at mediaeval 2015. In: MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany
- Boididou C, Papadopoulos S, Dang-Nguyen DT, Boato G, Kompatsiaris Y (2015) The cert-hunitn participation @ verifying multimedia use 2015. In: MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany
- Boididou C, Papadopoulos S, Dang-Nguyen D, Boato G, Riegler M, Middleton SE, Petlund A, Kompatsiaris Y (2016) Verifying multimedia use at mediaeval 2016. In: Working Notes Proceedings of the MediaEval 2016 Workshop, Hilversum, The Netherlands, October 20-21, 2016.
- Boididou C, Papadopoulos S, Middleton SE, Dang-Nguyen D, Riegler M, Petlund A, Kompatsiaris Y (2016) The VMU participation @ verifying multimedia use 2016. In: Working Notes Proceedings of the MediaEval 2016 Workshop, The Netherlands, Oct 20-21, 2016.
- Boididou C, Middleton SE, Jin Z, Papadopoulos S, Dang-Nguyen DT, Boato G, Kompatsiaris Y (2017) Verifying information with multimedia content on twitter. Multimedia Tools and Applications DOI 10.1007/s11042-017-5132-9
- Boididou C, Papadopoulos S, Apostolidis L, Kompatsiaris Y (2017) Learning to detect misleading content on twitter. In: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, ACM, ICMR '17, pp 278–286
- Cao J, Jin Z, Zhang Y (2016) Mcg-ict at mediaeval 2016 verifying tweets from both text and visual content. In: Working Notes Proceedings of the MediaEval 2016 Workshop, Hilversum, The Netherlands, October 20-21, 2016, CEUR-WS.org, vol 1739
- Castillo C, Mendoza M, Poblete B (2011) Information credibility on twitter. In: Proceedings of the 20th international conference on World wide web, ACM, pp 675–684
- Gupta A, Lamba H, Kumaraguru P, Joshi A (2013) Faking sandy: characterizing and identifying fake images on twitter during hurricane sandy. In: Proceedings of the 22nd international conference on World Wide Web companion, pp 729–736
- Gupta A, Kumaraguru P, Castillo C, Meier P (2014) Tweetcred: A real-time web-based system for assessing credibility of content on twitter. In: Proc. 6th International Conference on Social Informatics (SocInfo)

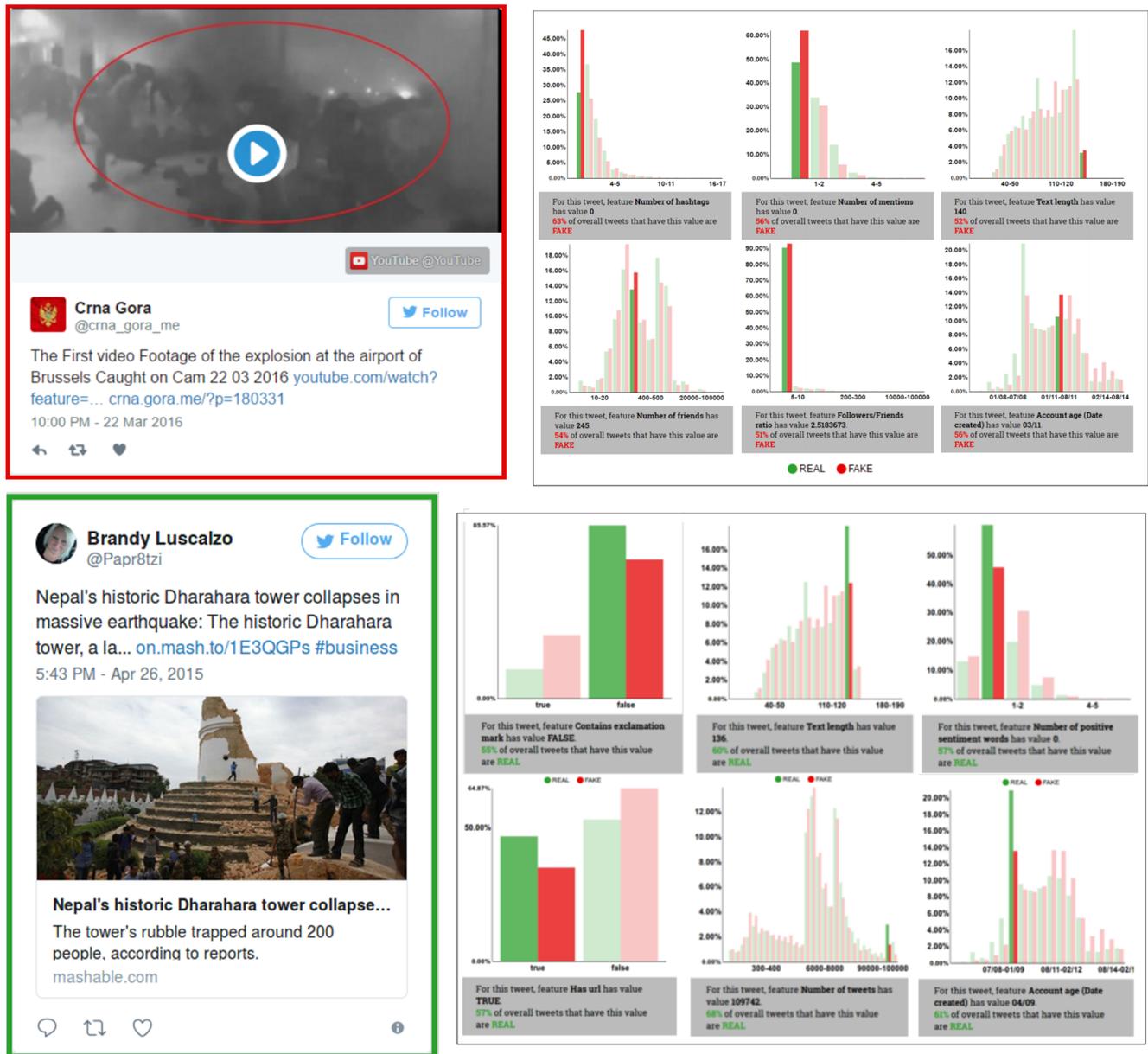


Fig. 6: Two cases of successful tweet classification and visual representation of feature distributions produced by the Tweet Verification Assistant.

12. Hassan N, Adair B, Hamilton J, Li C, Tremayne M, Yang J, Yu C (2015) The quest to automate fact-checking. In: Proceedings of the 2015 Computation and Journalism Symposium, pp 1–5
13. Jin Z, Cao J, Zhang Y, Zhang Y (2015) Mcg-ict at mediaeval 2015: Verifying multimedia use with a two-level classification model. In: MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany
14. Jin Z, Cao J, Zhang Y, Zhou J, Tian Q (2017) Novel visual and statistical image features for microblogs news verification. IEEE Trans Multimedia 19(3):598–608
15. Kanske P, Kotz SA (2010) Leipzig affective norms for german: A reliability study. Behavior research methods 42(4):987–991
16. Klein D, Manning CD (2003) Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1, Association for Computational Linguistics, ACL '03, pp 423–430
17. Kumar S, West R, Leskovec J (2016) Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. In: Proceedings of the

- 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016, ACM, pp 591–602
18. Maigrot C, Claveau V, Kijak E, Sicre R (2016) Mediaeval 2016: A multimodal system for the verifying multimedia use task. In: Working Notes Proceedings of the MediaEval 2016 Workshop, Hilversum, The Netherlands, October 20-21, 2016, CEUR-WS.org, vol 1739
 19. Martin N, Comm B (2014) Information verification in the age of digital journalism. In: Special Libraries Association Annual Conference, Vancouver
 20. Metaxas P, Finn S, Mustafaraj E (2015) Using twittertrails.com to investigate rumor propagation. In: Proceedings of the 18th ACM Conference Companion on Computer Supported Cooperative Work & Social Computing, ACM, pp 69–72
 21. Middleton S (2015) Extracting attributed verification and debunking reports from social media: Mediaeval-2015 trust and credibility analysis of image and video. In: MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany
 22. O'Donovan J, Kang B, Meyer G, Hollerer T, Adalii S (2012) Credibility in context: An analysis of feature distributions in twitter. In: 2012 International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2012 International Conference on Social Computing (SocialCom), IEEE, pp 293–301
 23. Oikawa MA, Dias Z, de Rezende Rocha A, Goldenstein S (2016) Manifold learning and spectral clustering for image phylogeny forests. *IEEE Trans Information Forensics and Security* 11(1):5–18
 24. Pandey RC, Singh SK, Shukla KK (2016) Passive forensics in image and video using noise features: A review. *Digital Investigation* 19:1–28, URL <http://dx.doi.org/10.1016/j.diin.2016.08.002>
 25. Phan QT, Budroni A, Pasquini C, Natale FGBD (2016) A hybrid approach for multimedia use verification. In: Working Notes Proceedings of the MediaEval 2016 Workshop, Hilversum, The Netherlands, October 20-21, 2016, CEUR-WS.org, vol 1739
 26. Ratkiewicz J, Conover M, Meiss M, Gonçalves B, Patil S, Flammini A, Menczer F (2011) Truthy: mapping the spread of astroturf in microblog streams. In: Proceedings of the 20th international conference companion on World wide web, ACM, pp 249–252
 27. Redondo J, Fraga I, Padrón I, Comesaña M (2007) The spanish adaptation of anew (affective norms for english words). *Behavior research methods* 39(3):600–605
 28. Resnick P, Carton S, Park S, Shen Y, Zeffer N (2014) Rumorlens: A system for analyzing the impact of rumors and corrections in social media. In: Proc. Computational Journalism Conference
 29. Rubin VL, Conroy NJ, Chen Y, Cornwell S (2016) Fake news or truth? using satirical cues to detect potentially misleading news. In: Proceedings of NAACL-HLT, pp 7–17
 30. Shao C, Ciampaglia GL, Flammini A, Menczer F (2016) Hoaxy: A platform for tracking online misinformation. In: Proceedings of the 25th Intern. Conf. Companion on World Wide Web, pp 745–750
 31. Silva E, de Carvalho TJ, Ferreira A, Rocha A (2015) Going deeper into copy-move forgery detection: Exploring image telltales via multi-scale analysis and voting processes. *J Visual Communication and Image Representation* 29:16–32
 32. Silverman C (2013) Verification handbook. The European Journalism Centre (EJC)
 33. Spyromitros-Xioufis E, Papadopoulos S, Kompatsiaris I, Tsoumakas G, Vlahavas I (2014) A comprehensive study over VLAD and Product Quantization in large-scale image retrieval. *IEEE Transactions on Multimedia* 16(6):1713–1728
 34. Sun S, Liu H, He J, Du X (2013) Detecting event rumors on sina weibo automatically. In: Web Technologies and Applications - 15th Asia-Pacific Web Conference, APWeb 2013, Sydney, Australia, April 4-6, 2013. Proceedings, Springer, Lecture Notes in Computer Science, vol 7808, pp 120–131
 35. Teyssou D, Leung JM, Apostolidis E, Apostolidis K, Papadopoulos S, Zampoglou M, Papadopoulou O, Mezaris V (2017) The invid plug-in: Web video verification on the browser. In: Proceedings of the 1st Workshop on Multimedia Verification
 36. Tsakalidis A, Papadopoulos S, Kompatsiaris I (2014) An ensemble model for cross-domain polarity classification on twitter. In: Web Information Systems Engineering–WISE 2014, Springer, pp 168–177
 37. Volkova S, Shaffer K, Jang JY, Hodas N (2017) Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, vol 2, pp 647–653
 38. Vosoughi S, Mohsenvand M, Roy D (2017) Rumor gauge: Predicting the veracity of rumors on twitter. *ACM Transactions on Knowledge Discovery from Data* 11:1–36
 39. Wu K, Yang S, Zhu KQ (2015) False rumors detection on sina weibo by propagation structures. In: 31st IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April

- 13-17, 2015, IEEE Computer Society, pp 651–662
40. Zampoglou M, Papadopoulos S, Kompatsiaris Y (2015) Detecting image splicing in the wild (web). In: IEEE International Conference on Multimedia & Expo Workshops (ICMEW), IEEE, pp 1–6
41. Zampoglou M, Papadopoulos S, Kompatsiaris Y, Bouwmeester R, Spangenberg J (2016) Web and social media image forensics for news professionals. In: Social Media in the Newsroom, Papers from the 2016 ICWSM Workshop, Cologne, Germany, May 17, 2016, AAAI Press, vol WS-16-19
42. Zampoglou M, Papadopoulos S, Kompatsiaris Y (2017) A large-scale evaluation of splicing localization algorithms for web images. *Multimedia Tools and Applications* 76(4):4801–4834
43. Zubiaga A, Aker A, Bontcheva K, Liakata M, Procter R (2017) Detection and resolution of rumours in social media: A survey. CoRR abs/1704.00656, URL <http://arxiv.org/abs/1704.00656>