Real-time Alert Framework for Fire Incidents Using Multimodal Event Detection on Social Media Streams

Thomas Papadimos*
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
papadimos@iti.gr

Nick Pantelidis
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
pantelidisnikos@iti.gr

Stelios Andreadis
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
andreadisst@iti.gr

Aristeidis Bozas
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
arbozas@iti.gr

Ilias Gialampoukidis
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
heliasgj@iti.gr

Stefanos Vrochidis
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
stefanos@iti.gr

Ioannis Kompatsiaris
Centre for Research & Technology Hellas
Information Technologies Institute
Thessaloniki, Greece
ikom@iti.gr

ABSTRACT

The frequency of wildfires is growing day by day due to vastly climate changes. Forest fires can have a severe impact on human lives and the environment, which can be minimised if the population has early and accurate warning mechanisms. To date, social media are able to contribute to early warning with the additional, crowd-sourced information they can provide to the emergency response workers during a crisis event. Nevertheless, the detection of real-world fire incidents using social media data, while filtering out the unavoidable noise, remains a challenging task. In this paper, we present an alert framework for the real-time detection of fire events and we propose a novel multimodal event detection model, which fuses both probabilistic and graph methodologies and is evaluated on the largest fires in Spain during 2019.

Keywords

Alert framework, social media, event detection, kernel density estimation, community detection.

*corresponding author

INTRODUCTION

Climate change has largely increased the likelihood that fires will start more often and burn more intensely and widely than they have in the past. In fact, circa 2.5 million acres were burned in the US during the fire season of 2020 and it was considered the most destructive season on record, while more than 200 wildfires affected western and southern Turkey in 2021, which is more than eight times the average between 2008 and 2020[1]. When huge swaths of forest burn, it is hard for forests to self-generate and the wildfires can lead to loss of property, crops, resources, animals and human lives, as well as to the deterioration of the air quality. According to the World Health Organization[2], 6.2 million people were affected by wildfires between 1998-2017 with 2,400 deaths worldwide. Moreover, a large quantity of carbon dioxide and carbon monoxide were released, which causes a range of health issues, including respiratory and cardiovascular problems. Forest fires can have a significant impact on mortality and morbidity depending on their size, their speed, and whether the population has a timely warning to evacuate.

In the last decade social media platforms have been proven to be a valuable sources of information for early warning tools, considering the rich and prompt information they relay (Merchant et al. 2011; Reuter et al. 2018; Panagiotopoulos et al. 2016). When major events occur, people usually turn to social media to share what they have witnessed, document what is happening in real time, and exchange key information. Twitter is one of the most well-known social media platforms, used by 330 million users around the world, with 6,000 and 200 billion tweets posted every second and every year respectively[3]. There are even cases where an event was reported on Twitter before it was reported on the news network (Pandhare and Shah 2017). Regarding fire emergency situations that take place in a particular location or area, Twitter's data can be very useful to civilians and rescue teams for immediate crisis response. However, the large streams of published posts carry lots of noise, e.g. because of the metaphorical use of incident-related words, causing possible false alerts.

This paper presents an alert framework that detects events in real time from Twitter for early warning applications that can assist first responders in decision making. In this particular research, we are focusing on fire events and the Spanish language, but the framework could easily be adapted to other disasters and languages. The framework consists of three steps: the acquisition of tweets related to fires, the detection of events and the generation of insights towards producing an alert. Furthermore, we propose a novel multimodal model that fuses probabilistic and graph methodologies in order to extract events from the collected tweets. In a nutshell, the contribution of our work is:

- A real-time alert framework using Twitter information for early warning and decision making.
- A novel multimodal event detection methodology that overcomes the limitations of the baseline techniques.

The paper is organised as described below. In the next section, we provide a research on related works, compared to the proposed. Following, the proposed framework is presented, along with the event detection methodology and the necessary background for the models that are fused. Next, the evaluation of the event detection model and the respective results are described, while the last section concludes with future work.

RELATED WORK

Social media have become extremely popular the last years, with millions of unique users and millions of posts every day. A tremendous amount of information is generated in real time by the constant interaction between the users. Real-life events and incidents are mentioned in social media every day. Uncovering these events automatically has attracted the interest of the scientific community and a variety of techniques have been introduced. Event detection has real-world application, such as crisis management and decision making, as correct decisions can be made by knowing what is happening, where it is happening and who is involved.

Goswami and Kumar 2016 provide a survey on event detection in online social text stream data, such as newswires, web forums, blogs, emails and micro-blogs. This survey classifies the event detection based on four event dimensions: (i) Thematic, (ii) Temporal, (iii) Spatial, and (iv) Network structure. Furthermore, some available tools for event detection are presented and described.

More recently, Saeed et al. 2019 prepared a survey on event detection techniques on Twitter streams. This survey also proposed a general Event Detection on Twitter (EDoT) framework that consists of (i) data acquisition, (ii) feature extraction, (iii) event detection method, and (iv) event representation. In addition, various data collection techniques, evaluation metrics and benchmark methodologies were summarised and compared.

[1] https://www.bbc.com/news/58159451
[2] https://www.who.int/health-topics/
[3] https://www.internetlivestats.com/twitter-statistics/

Focusing on a natural disaster, Bruijn et al. 2019 presented an algorithm to detect floods through tweets, along with the Twitter dataset they used in their experiments. An algorithm based on the SEQAVG burst detection method was employed to detect a burst on each incoming tweet about the flood. They expanded the algorithm so as to take into consideration the fluctuations in the number of Twitter users according to the time of day and used a more dynamic threshold for the detection of bursts.

Additionally, Burel et al. 2017 introduced a semantically enhanced Dual-Convolutional Neural Network deep-learning model in order to identify events that arise during crisis situations on Twitter. The proposed model is an extension of an existing word-embedding CNN model with an additional semantic representation layer, created with named-entities in Twitter together with their semantic concepts.

Suma et al. 2017 used big data and machine learning techniques to analyse Twitter data related to traffic incidents about London with the aim of detecting the location and time of an event happening in the city. A classification model was trained by using logistic regression with stochastic gradient descent on tweets that were geolocated by Google Maps Geocoding API and classified into two categories, i.e., traffic-related and non-traffic related.

In another work, Zhang et al. 2018 employed deep learning in traffic-related Twitter data, but this time to detect traffic accidents in Northern Virginia and New York City. Two deep learning methods were investigated, i.e., Deep Belief Network (DBN) and Long Short-Term Memory (LSTM) and were implemented on extracted token features.

Li et al. 2017 introduced a framework to detect real-time events on Twitter. After splitting the tweet term space into semantic categories (classes), a semantic class-based event clustering algorithm was implemented to group the tweets talking about the same event into clusters. Also, a temporal information identification module was developed to filter out clusters that are referring to past events.

In a more recent study, Hasan et al. 2019 proposed another real-time event detection system from the Twitter data stream called TwitterNews+ Framework. The framework incorporates two modules: (i) a search module that searches for tweets by specialised inverted indices, and (ii) EventCluster Module, where the tweets are clustered with an incremental clustering approach to provide a low computational cost solution, to detect both major and minor newsworthy events in real time from the Twitter data stream.

Furthermore, Aldhaheri and J. Lee 2017 put forward a novel framework for detecting events on large social media. A Temporal Social Network Graphs Event Detection framework was implemented based on a temporal approach that transforms social media streams into temporal images and detects structural changes of the social network that reflects an occurrence of an event, which allows for building a better event detection predictive model.

Weng and B.-S. Lee 2011 introduced EDCoW (Event Detection with Clustering of Wavelet-based Signals) to detect real life events on Twitter by analyzing the text of the tweets. EDCoW transforms each term to signals by applying wavelet analysis to reduce space and storage, then the trivial terms are filtered out based on their auto-correlation signals, and finally the remaining term signals are clustered together with a modularity-based graph partitioning to group the event-related terms into events.

Shang D et al. 2018 proposed a multi-view clustering model that takes into consideration the topic and time-series information from tweets. A Latent Disctrict Allocation (LDA) model was applied to the text of the tweets to create a matrix with the probability of words under a topic. These topic words were selected as keywords to describe the event. Then, by implementing the Dynamic Time Warping (DTW) algorithm, the similarity of the above-mentioned keywords was calculated in a matrix. Finally, topical similarity and time-series similarity matrices were applied in a multi-view clustering model for event detection.

Regarding multimodal learning, the topic-modelling-based approaches for real-time event detection associate each tweet with a probability distribution over various latent topics to find the hidden semantic structures from a collection of tweets used to guide the event detection task. These methods rely on sophisticated models to infer latent topics.

Madani et al. 2015 applied a nonparametric Bayesian model, called Hierarchical Dirichlet Processes (HDP) (Teh et al. 2006), to detect trending topics from the Twitter data stream. For this method, a vector of topics is initially discovered by applying the generative model of HDP on tweets so that, for each tweet, the distribution of topics is calculated by exploiting the vector of topics. The topic with the highest probability in the distribution of topics is considered a trending topic.

Spatio-Temporal Multimodal TwitterLDA (STM-TwitterLDA) (Cai et al. 2015) is a topic-model-based framework for event detection that extracts text, image, location, timestamp and hashtag-based Twitter features from each tweet in the Twitter data stream as input and then jointly models the probability distribution of these features to detect events. Note that these features, apart from plain text, may not always be present in a tweet. STM-TwitterLDA employs a Support Vector Machine (SVM) classifier to remove noisy images, a latent filter to remove generic images

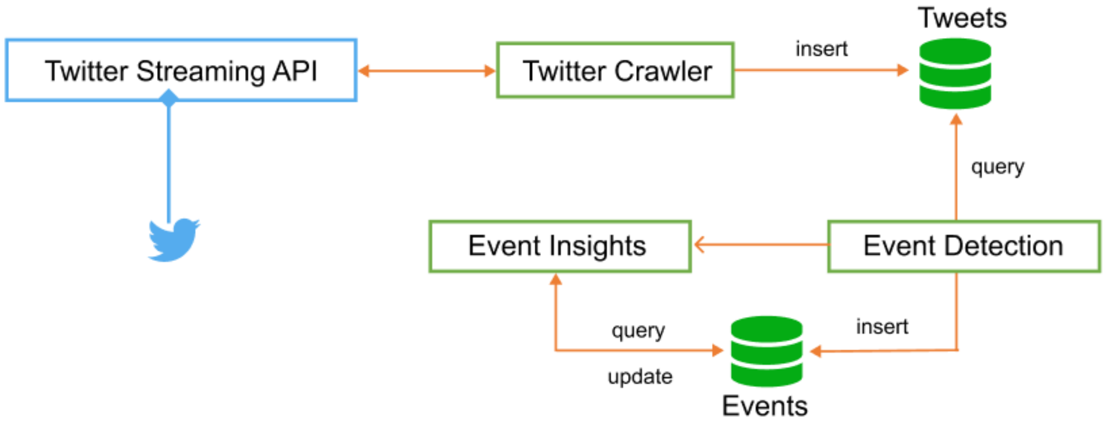

Figure 1. Overall architecture of the framework

and words, and Convolutional Neural Networks (CNNs) (Jia et al. 2014) to extract visual features from images to leverage in event detection. Finally, maximum-weighted bipartite graph-matching is applied in consecutive periods to track the evolution of the detected events.

In contrast to the above works, the novel event detection methodology that we propose in this paper does not examine the fluctuation of tweets collected per day or the relevance of the content, nor focuses solely on one characteristic (e.g. spatiotemporal info, social network, etc.), but fuses two different methods, taking into consideration both the density of posted tweets and the structure of the users community, which are found to be altered during an incident. In this way, the detection of events is not exclusively depended on high activity on Twitter, which is not always the case, especially for forest fires.

THE PROPOSED FRAMEWORK

The overall architecture of our framework is illustrated in Figure 1. First, we collect the tweets that will be used in order to detect events. To retrieve them we have developed the Twitter Crawler that calls the Twitter Streaming API. When a tweet matches the predefined search criteria, it is stored in a MongoDB database named "Tweets". The Event Detection module, running in a time interval that can be set, reads the tweets that have been stored in the database and applies the proposed event detection methodology. When the occurrence of an event is identified, the timestamp and the group of tweets it comprises are inserted in a MongoDB database named "Events". When the Event Detection module concludes, the Event Insights module is triggered and for each event the top ten most mentioned keywords are extracted and stored, in order to provide the user with more details, and a corresponding alert is generated.

The underlying parts of the framework are described in detail in the following subsections.

Social media data acquisition

In order to collect tweets from Twitter, we use the Hosebird Client[4], a Java HTTP client for consuming Twitter's standard Streaming API. This API gives access to the real-time stream of public tweets. It allows the specification of keywords based on which the API will filter the new tweets and return only the subset that matches these criteria. Since the focus of this particular implementation has been decided to be on fires in Spain, the keywords used to collect fire-related tweets in Spanish were the following: *incendio* (fire), *llamasdefuego* (flames of fire) and *bomberos* (firefighters). When a new tweet is retrieved, we check that its text is indeed in Spanish and then we store it in a MongoDB database.

At this point we should highlight some limitations that are inherent in the acquisition and usage of social media data for detecting disaster-related events. One limitation is that retrieval is based on fixed search terms. Having very specific keywords leads to a better monitoring of particular phase of a disaster, but risks matching sufficient tweets. On the contrary, setting more generic criteria can result to an immense number of matched tweets that refer to many different aspects of a disaster (e.g. type or phase) and then categorization techniques would be imminent. Since this

[4] https://github.com/twitter/hbc

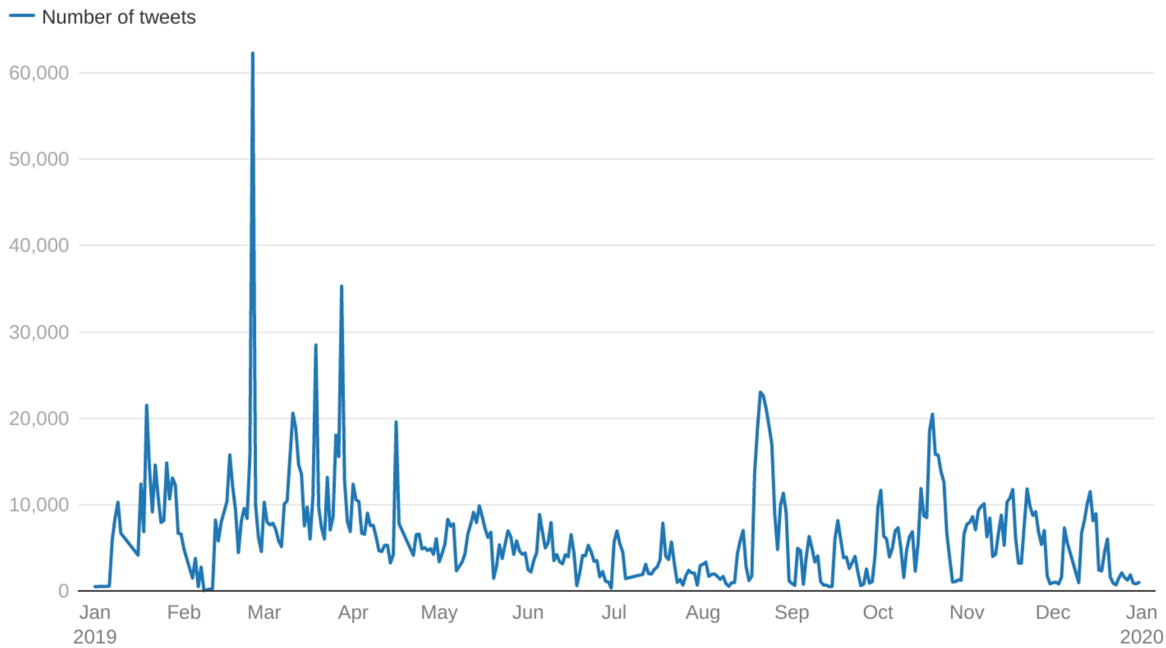

Figure 2. Daily consumption of tweets that are related to fires in Spain during 2019

work does not focus on a particular aspect, but rather on fire events in general, we have preferred more generic keywords, as mentioned above.

Another limitation is that social media data often lack geoinformation (some platforms provide no geoinformation at all) or their position is different to the location of the incident mentioned in their text. On the other hand, analyzing the text in order to locate the event is language-dependent and this can also present some weaknesses. In our case, for instance, the Spanish language is spoken by millions of people around the world and, as a result, we collect many tweets that refer to fires, but not in the country of Spain, which is our focus in this work. To filter the tweets accordingly, we have used a list of place names in Spain[5] and kept only the tweets whose text contained at least one of these names.

To provide the reader with an indication of the size of the collection, for the year of 2019 (which is also the time frame of the evaluation in the next section) the number of the retrieved tweets concerning fires in Spain was 2,205,946 tweets. Moreover, Figure 2 illustrates the number of tweets collected per day. It is also visible in this figure that during the summer season, which traditionally is a period of numerous fires, the activity on Twitter was low, indicating the need for an event detection methodology that does not rely on peaks on the fluctuation of collected tweets.

Event detection

Background

The proposed event detection methodology considers two modalities: the first one refers to Kernel Density Estimation (Efron et al. 2014) and the second one to Community Detection (De Meo et al. 2011). To ensure that the reader is familiarised with the two modalities, some basic information is given here.

Kernel Density Estimation (KDE) aims to detect events, considering not only the number of tweets in a specific time period, but also the sparsity and the density in the time frame where they were posted. Thus, for each day there is a specific density score (f_T) based on the number of posted tweets and their publication time. Let $(x_1, x_2, ..., x_n)$ be a sample of tweets timestamps (in hours) in a day, from some distribution with an unknown density f. The main objective is to estimate the unknown probability density function $f(x)$. To tackle that, a histogram can be used that is simple to understand and works reasonably well. However, to reduce the arbitrary placement of the histogram bins, we centre a box function K on each data point x_i and average those functions to obtain a probability density function.

[5]http://download.geonames.org/export/dump/

This is a simple kernel density estimator:

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} K(x - x_i) \tag{1}$$

where K is a box function and x the last timestamp of the day (in hours). Since each function K has $\int K dx = 1$, we divide by h to ensure that $\int f(x)dx = 1$. Finally, the kernel density estimator is:

$$f(x) = \frac{1}{Nh}\sum_{i=0}^{N} K(\frac{x - x_i}{h}) \tag{2}$$

where K is a symmetric, but not necessarily positive function that integrates to 1 and $h > 0$ is the bandwidth. Though many kernel K functions are viable, we use the common Gaussian distribution, such that:

$$K(\frac{x - y}{h}) = N(\frac{x - y}{h}, 0, h) \tag{3}$$

where N is the normal Density. The Gaussian distribution has been chosen for two reasons. First, it gives a ready plug-in value for the optimal bandwidth h and, secondly, we have discovered through experiments that the choice of kernels has almost no effect on the performance of our method. On the other hand, the choice of bandwidth is important and we rely on (Sheather and Jones 1991).

In parallel, the second modality, namely the Community Detection (CD), considers a method that exploits graphs and is used in machine learning to divide the network of social media users into communities with similar properties, e.g. discussing the same topic.

The first step is to denote by $G(N, L)$ the social network, with N nodes that each represents a Twitter user account and L links, where a link between two users (i, k) exists if a user n_i mentions or is mentioned by another user n_k. This particular relationship has been preferred over others (e.g. following), because it expresses a more temporary connection between users, and user communities shift continuously based on trending topics and events.

The bibliography suggests several community detection algorithms, such as Edge Betweenness (Newman and Girvan 2004), Fast Greedy (Clauset et al. 2004), Label Propagation (Raghavan et al. 2007), Louvain (De Meo et al. 2011), Walktrap (Pons and Latapy 2005), and Infomap (Rosvall et al. 2009; Bohlin et al. 2014). After comparing the above approaches in terms of different metrics, the Louvain algorithm has been selected as the most fast in large networks. The outcome of the algorithm is the number of detected communities and the set of nodes that belong to each community.

Modularity is a metric with regards to the structure of networks and it was designed to measure the strength of the division of a network into modules, i.e. communities. Networks with high modularity have dense connections between the nodes within modules, but sparse connections between nodes in different modules. The calculation of modularity follows:

$$MS = \frac{1}{2m}\sum_{i=1}^{c}(e_{ii} - \alpha_i^2) \tag{4}$$

where m is the number of edges, e_{ij} is the fraction of links between a node in community i and a node in community j, and a_i is the fraction of links between two members of the community.

Methodology

In this section, we describe the event detection methodology, which consists of four parts. First, tweets are obtained with a query to the database where they are stored for a specific time period $T \in (t1, t2, ..., tn)$. Next, the timestamps of the tweets are used by the Kernel Density Estimation for the calculation of the Density Score (DS). In parallel, the Modularity Score (MS) is estimated by the Community Detection. Finally, the two scores are combined by a fusion method that estimates whether an event is happening in that period. The architecture of the methodology can also be seen in Figure 3.

Considering a query in interval $I = (a, c)$, where $I \cap T \neq \emptyset$, and an event in day $b \in I$, DS is calculated for each day (a, b, c) by comparing the DS of the current to the past date. Particularly, DS for day b takes into account the

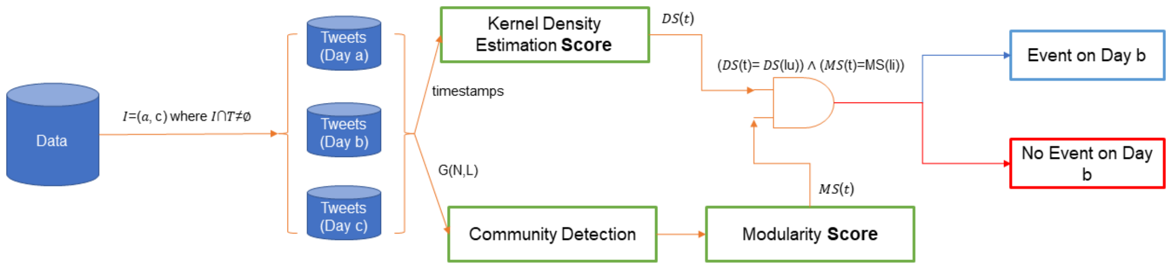

Figure 3. Architecture of the proposed event detection methodology

timestamps from day a to day b, since in some occasions the event happens slightly during the change of day, which means that the gap on 23:59 could be the point where the tweets present the highest density in time. To tackle this, the timestamps are converted to hours and x in Equation (2) concerns the last timestamp, which is probably in day b. We consider the maximum density of Equation (2) as the DS feature for each day in I. Finally, in the time frame I, a local maximum in day b with $DS > 0.12$ concerns a detected event, with the threshold having been defined as the best performing for detecting events with the standalone KDE (described in Section Results).

Although KDE is an efficient method for detecting events in social media traffic, the noise of such data can still lead to false peaks. Thus, we propose an additional feature for calibrating KDE. This calibration corresponds to the changes in the Twitter user communities during an event in the time period I, where MS (Equation 4) is calculated for each day separately and corresponds to four categories. The first one is the negative modularity ($MS < 0$), where no communities are formulated. The second is the sub-optimal partition, in which there are a few correlations between the communities ($0.2 < MS < 0.4$), which may discuss the same topic (event), but are structured separately. The third category is the optimal partition, in which there are no correlations between the communities ($MS > 4$). Different communities in our case means different clusters of Twitter users and each cluster corresponds to a specific topic. The last category is the single community with $MS = 0$. In this case, all user accounts discuss the same topic, which is not realistic for the large streams of tweets and, hence, user mentions that we examine. Thus, we have concluded to consider an event when $MS > 0.2$.

In order to combine the aforementioned scores DS and MS, in a time period $I = (a, c)$ where $I \cap T \neq \emptyset$, $lu \in I$ is the local maximum, $li \in I$ is the local minimum, and $DS(t)$, $MS(t)$ are the corresponding time series score functions, the fusion method detects an event according to the following equation:

$$Event = \begin{cases} TRUE & DS(lu) \wedge MS(li) \\ FALSE & \end{cases} \tag{5}$$

where $DS(lu) > 0.12$ and $MS(li) > 0.2$.

It should be noted here that the methodology presented above checks for an event in a daily basis, due to the fact that it is easier in this way to be evaluated (Section Evaluation), since real disasters (ground truth) are usually recorded by date and not by hour. Nevertheless, the method can be easily adjusted to support hourly detection, which is more useful to the first responders, who need to be alerted as soon as possible.

Providing insights

After an event is detected by the aforementioned methodology, the outcome is the timestamp of the event and the group of tweets it comprises. Aiming to assist the user, e.g. the first responder, to understand more about the potential incident, the next step is to extract some keywords regarding the event. To this end, the text of the comprised tweets is retrieved, the Spanish stop words[6] are removed and the top ten most mentioned words are calculated. This list can facilitate the end user to comprehend at a glance what the context of an event is, instead of examining the multitude of the individual tweets.

We should highlight that this part of the framework has been designed in such a manner that it can be easily extended with further machine learning techniques that could offer more insights, such as georeferencing to estimate the location of the event or Natural Language Processing (NLP) to autogenerate an alert message.

[6] https://github.com/Alir3z4/stop-words/blob/master/spanish.txt

Table 1. Largest fires in Spain in 2019

Description	*Date*
Forest Fire in Canary Islands	2019-05-15
Forest Fire in Andalusia	2019-06-01
Fire in Torre de l'Espanyol	2019-06-26
Wildfire in Community of Madrid	2019-06-28
Forest Fire in Castile and León	2019-06-28
Forest fire in Sierra de Gador	2019-07-13
Fires in Zaragoza	2019-07-23
Forest fire in Segovia	2019-08-04
Forest fire in Artenara	2019-08-10
Forest fire in Cazadores	2019-08-12
Forest fire in Valleseco, Gran Canaria	2019-08-17

Table 2. Evaluation results in terms of accuracy

Method	*Accuracy*	*F1-score*	*Threshold*
STA/LTA	0.7726	0.0235	$R > 1.5$
Z-Score	0.8301	0.0606	$z - score > 1$
KDE	0.8986	0.1395	$DS > 0.12$
KDE+CD	**0.9589**	0.2857	$DS > 0.12, MS > 0.2$

EVALUATION

Ground truth

In order to be able to evaluate our proposed event detection methodology, i.e. in terms of accuracy, we require some ground truth, which in this case refers to a list of known fire events. The focus of the evaluation experiments has been selected to be the detection of fire incidents that occurred in Spain during the year of 2019. According to Internal Displacement Monitoring Center[7] 10,717 fires were reported in Spain during 2019, 3,544 of which burned more than one hectare. Nevertheless, it was not possible to discover all these fires and we considered 11 of the largest fires of that year, as reported by the Copernicus Emergency Management Service[8], shown in Table 1.

Results

The performance of our methodology is compared to two well-established event detection algorithms, namely Z-Score (Shiffler 1988) and STA/LTA (Allen 1978). The first one is a statistical metric that indicates how many standard deviations away a given observation is from the mean, assuming a Gaussian distribution, and takes two parameters: mean and standard deviation. The latter is a method that comes from the field of Geophysics and specifically from earthquakes detection. For each point, the R or STA/LTA ratio is calculated, where STA is the N-s-point short-term average and LTA is the N-L-point long-term average. In contrast to KDE, which is a method that concerns the temporal density of the posted tweets, both methods consider only the fluctuation of the number of tweets per day.

The performance regards the ability of each method to detect real fire events in the aforementioned dataset. This ability is measured with accuracy and F1-score. Since different thresholds can be set for all compared methods, above which an event is considered, we have selected the thresholds achieving the best performance per method. In addition to the baselines and the proposed fusion methodology, we also include in the comparison the standalone KDE with best accuracy (0.8986) with $DS > 0.12$. The results can be seen in Table 2, where KDE+CD clearly outperforms the other techniques, with an accuracy of 0.9589, which is 23.8% over STA/LTA and of 15.2% over Z-score. Evidently, CD adds significant value to the fusion method due to its ability to detect additional real events, as described in the following paragraphs.

Furthermore, we present cross-validation for comparing the performances in a specific time period, i.e. between 3 and 14 August 2019, where three real fire incidents happened on 04/08, 10/08, and 12/08. Figure 4 displays the

[7]https://www.internal-displacement.org/sites/default/files/inline-files/GRID-2019-Disasters-Figure%2DAnalysis-DWildfires-Spain.pdf
[8]https://emergency.copernicus.eu/mapping/list-of-activations-rapid

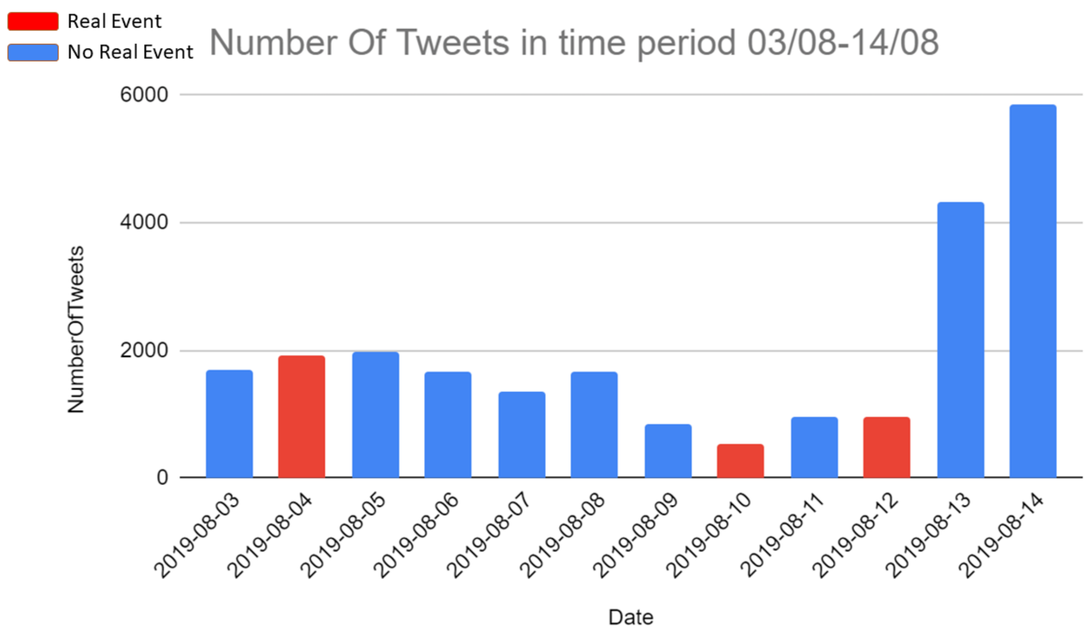

Figure 4. Number of collected tweets between 03/08 and 14/08, where red indicates the existence of a real event

number of tweets posted each day and red bars indicate the date of a real event, while Figure 5 shows the scores of each method per day and the detected/non-detected events.

In detail, red points in Figure 5 correspond to real events, listed also in Table 1, and blue points refer to days without events. The yellow cross sign means that the event was detected, while the black X sign means that it was not. For instance, STA/LTA (top left) did not detect the event on 04/08 (red and X) and it falsely detected an event on 14/08 (blue and cross). Additionally, the green line signifies the threshold for each method (mentioned in Table 2) and the grey arrows refer to the proposed fusion that managed to detect two of the three events in that period. It is evident that STA/LTA and Z-score show a similar pattern in their scores, as they are affected by the fluctuation of tweets per day, while CD and KDE differentiate by taking into account the temporal density and the structure of the social networks, achieving in this way a better performance. It is also worth-mentioning that KDE falsely detected an event on 07/08 that KDE+CD disregarded due to a lower MS, thus proved that the fusion with CD is valuable to the detection.

Finally, to showcase the actual outcome of the proposed methodology as well as the contribution of the Event Insights module, we list in Table 3 all the dates during 2019 where an event has been detected by KDE+CD, along with the top ten most mentioned keywords per event (also translated in English, when needed). The bold keywords refer to place names, which can be the basis for identifying the locations of the detected events. As it can be seen in the third column, we provide the reader with a short description of each incident and a link to a relevant online article. It is notable that no incident is completely irrelevant, but not all of them refer to forest fires or wildfires and not all of them concern Spain.

[9] https://www.thelocal.es/20190126/discovery-of-julens-lifeless-body-brings-tragic-end-to-rescue-mission/
[10] https://borgenproject.org/electricity-in-venezuela/
[11] https://www.cooperativa.cl/noticias/pais/policial/incendios/incendio-afecto-al-segundo-piso-del-serviu%2en-santiago/2019-03-23/090227.html
[12] https://www.aa.com.tr/en/world/fire-kills-at-least-20-in-peru-passenger-bus/1439361
[13] https://gfmc.online/media/2019/04-2019/news_20190402_co.html
[14] https://english.elpais.com/elpais/2019/06/28/inenglish/1561707211_884034.html
[15] https://www.bbc.com/news/world-europe-49224776
[16] https://www.eumetsat.int/gran-canaria-wildfires
[17] https://english.elpais.com/elpais/2019/08/13/inenglish/1565688485_741200.html
[18] https://www.clarin.com/sociedad/flores-nene-3-anos-murio-incendio-hotel-familiar_0_CNHs_rSC.html
[19] https://www.telesurenglish.net/news/Breaking-Quito-Militarized-Under-Curfew-20191012-0007.html
[20] https://crisis24.garda.com/insights-intelligence/intelligence/risk-alerts/amhnn2vhu2a2fwdhp/spain-clashes-reported-in-barcelona-october-15-as-thousands-protest-jailing-of-separatists

Table 3. Insights for all events detected by KDE+CD in 2019

Date	*Top ten most mentioned keywords (with translation)*	*Description*
2019-01-26	incendio (fire), bomberos (firefighters), forestal (forest), mineros (miners), julen, vásquez, gracias (thank you), mujer (woman), santuario (sanctuary), cuerpo (body)	Spanish miners find body of boy named Julen trapped in borehole[9]
2019-03-10	incendio (fire), fuentes (sources), san (saint), líneas (lines), corpoelec, **guri**, afectó (affected), vegetación (vegetation), tarde (late), vinculadas (linked)	Blackout in Venezouela with possible cause a fire at the Guri Dam[10]
2019-03-23	incendio (fire), conagua, edificio (edifice), bomberos (firefighters), insurgentes (insurgents), sur (south), registra (records), cdmx, eje (axis), controlado (checked)	Fire on the second floor of the Serviu building in Santiago[11]
2019-04-01	incendio (fire), bomberos (firefighters), fiori (flower), muertos (dead), bus, **lima**, personas (persons), terminal, forestal (forest), empresa (business)	Fire kills at least 20 in Peru passenger bus[12]
2019-04-03	incendio (fire), bomberos (firefighters), parque (park), arauco, mall, voluntarios (volunteer), mascotas (pets), tienda (store), evacuados (evacuees), **santiago**	Forest fire in the Los Katíos National Natural Park[13]
2019-07-01	incendio (fire), **tarragona**, sándwich, lonchas (slices), ume, chorizo, comida (meal), militares (military), bombeross (firefighters), menú	Large wildfire that began in Tarragona province[14]
2019-08-04	incendio (fire), bomberos (firefighters), millones (millions), hectáreas (hectares), granja (farm), forestal (forest), **siberia**, febrero (february), toneladas (tons), día (day)	Wildfires in Siberia[15]
2019-08-10	incendio (fire), bomberos (firefighters), **artenara**, forestal (forest), **canaria**, medios (media), declarado (declared), **grancanaria**, cumbre (summit), zona (zone)	Gran Canaria wildfires[16]
2019-08-13	incendio (fire), bomberos (firefighters), **canaria**, forestal (forest), estabilizado (stabilized), **canarias**, **ifartenara**, **ifcazadores**, **ifgrancanaria**, brif (briff)	Stabilisation of Gran Canaria's fire[17]
2019-09-27	incendio (fire), bomberos (firefighters), cumbre (summit), flores (flowers), hotel, fuego (fire), años (years), hospital, forestal (forest), familias (families)	3-year-old boy died during a fire in a family hotel in Flores[18]
2019-10-12	incendio (fire), bomberos (firefighters), contraloría (comptroller), **quito**, general, personas (persons), edificio (edifice), registra (records), video, teleamazonas	Fire in a van of Teleamazonas television during protests in Equador[19]
2019-10-15	bomberos (firefighters), incendio (fire), nacional (national), disparando (shooting up), nivel (level), alto (high), policia (police), llegado (arrived), locura (craziness), **barcelona**	Fires during protest in Barcelona[20]

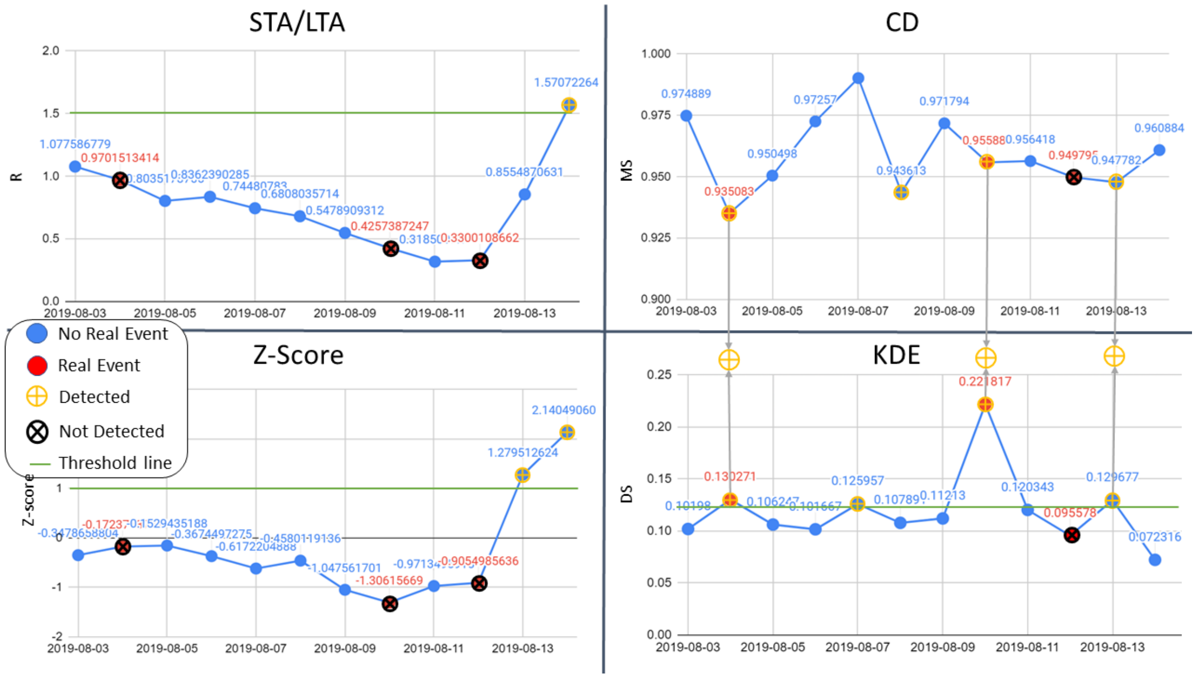

Figure 5. Scores and detection performance of each methodology between 03/08 and 14/08 (grey arrows refer to the fusion of KDE and CD)

CONCLUSION AND FUTURE WORK

In this paper we have presented a real-time alert framework for detecting fire events in Spain from Twitter, which can support first responders in disaster management and decision making. The framework also involves a novel multimodal event detection methodology that fuses two modalities. The first modality refers to Kernel Density Estimation that considers temporal sparsity of tweets and the second to Community Detection, a graph-based algorithm for grouping Twitter users in communities, which can then be measured with the Modularity Score. In the evaluation stage, the above method has been compared with traditional probabilistic and outlier algorithms for event detection, on a dataset of Spanish, fire-related tweets posted in 2019. The results have shown to superiority of the proposed KDE+CD, with an accuracy of 0.9589.

The experiments have also exposed a limitation of our framework. Examining the insights of the detected events, it is obvious that our attempt to narrow the tweets to the ones referring to Spain with a simple text matching is not enough and it reveals the need for a georeferencing technique that will add geoinformation to the posts in a more sophisticated manner. Furthermore, NLP methods could be exploited for the analysis of the detected events, so as to autogenerate alert messages that will describe the incidents in a more natural way, instead of merely listing the top keywords. Finally, future work could involve the evaluation of the proposed method for one-hour time frames (in addition to one-day) and the extension of the technique in order to include other modalities as well.

ACKNOWLEDGEMENTS

This work was supported by the INGENIOUS and CALLISTO projects. INGENIOUS has received funding from the European Union's Horizon 2020 Research & Innovation Programme and the Korean Government under Grant Agreement No. 833435. CALLISTO has received funding from the European Union's Horizon 2020 Research & Innovation Programme under Grant Agreement No. 101004152. This publication reflects only the authors' view and the Research Executive Office (REA) and the European Commission are not responsible for any use that may be made of the information it contains.

REFERENCES

Aldhaheri, A. and Lee, J. (2017). "Event detection on large social media using temporal analysis". In: *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, pp. 1–6.

Allen, R. V. (1978). "Automatic earthquake recognition and timing from single traces". In: *Bulletin of the seismological society of America* 68.5, pp. 1521–1532.

Bohlin, L., Edler, D., Lancichinetti, A., and Rosvall, M. (2014). "Community detection and visualization of networks with the map equation framework". In: *Measuring scholarly impact*. Springer, pp. 3–34.

Bruijn, J. A. de, Moel, H. de, Jongman, B., Ruiter, M. C. de, Wagemaker, J., and Aerts, J. C. (2019). "A global database of historic and real-time flood events based on social media". In: *Scientific data* 6.1, pp. 1–12.

Burel, G., Saif, H., Fernandez, M., and Alani, H. (2017). "On semantics and deep learning for event detection in crisis situations". In.

Cai, H., Yang, Y., Li, X., and Huang, Z. (2015). "What are popular: exploring twitter features for event detection, tracking and visualization". In: *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 89–98.

Clauset, A., Newman, M. E., and Moore, C. (2004). "Finding community structure in very large networks". In: *Physical review E* 70.6, p. 066111.

De Meo, P., Ferrara, E., Fiumara, G., and Provetti, A. (2011). "Generalized louvain method for community detection in large networks". In: *2011 11th international conference on intelligent systems design and applications*. IEEE, pp. 88–93.

Efron, M., Lin, J., He, J., and De Vries, A. (2014). "Temporal feedback for tweet search with non-parametric density estimation". In: *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pp. 33–42.

Goswami, A. and Kumar, A. (2016). "A survey of event detection techniques in online social networks". In: *Social Network Analysis and Mining* 6.1, pp. 1–25.

Hasan, M., Orgun, M. A., and Schwitter, R. (2019). "Real-time event detection from the Twitter data stream using the TwitterNews+ Framework". In: *Information Processing & Management* 56.3, pp. 1146–1165.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). "Caffe: Convolutional architecture for fast feature embedding". In: *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678.

Li, Q., Nourbakhsh, A., Shah, S., and Liu, X. (2017). "Real-time novel event detection from social media". In: *2017 IEEE 33Rd international conference on data engineering (ICDE)*. IEEE, pp. 1129–1139.

Madani, A., Boussaid, O., and Zegour, D. E. (2015). "Real-time trending topics detection and description from Twitter content". In: *Social Network Analysis and Mining* 5.1, pp. 1–13.

Merchant, R. M., Elmer, S., and Lurie, N. (2011). "Integrating social media into emergency-preparedness efforts". In: *New England journal of medicine* 365.4, pp. 289–291.

Newman, M. E. and Girvan, M. (2004). "Finding and evaluating community structure in networks". In: *Physical review E* 69.2, p. 026113.

Panagiotopoulos, P., Barnett, J., Bigdeli, A. Z., and Sams, S. (2016). "Social media in emergency management: Twitter as a tool for communicating risks to the public". In: *Technological Forecasting and Social Change* 111, pp. 86–96.

Pandhare, K. R. and Shah, M. A. (2017). "Real time road traffic event detection using Twitter and spark". In: *2017 International conference on inventive communication and computational technologies (ICICCT)*. IEEE, pp. 445–449.

Pons, P. and Latapy, M. (2005). "Computing communities in large networks using random walks". In: *International symposium on computer and information sciences*. Springer, pp. 284–293.

Raghavan, U. N., Albert, R., and Kumara, S. (2007). "Near linear time algorithm to detect community structures in large-scale networks". In: *Physical review E* 76.3, p. 036106.

Reuter, C., Hughes, A. L., and Kaufhold, M.-A. (2018). "Social media in crisis management: An evaluation and analysis of crisis informatics research". In: *International Journal of Human–Computer Interaction* 34.4, pp. 280–294.

Rosvall, M., Axelsson, D., and Bergstrom, C. T. (2009). "The map equation". In: *The European Physical Journal Special Topics* 178.1, pp. 13–23.

Saeed, Z., Abbasi, R. A., Maqbool, O., Sadaf, A., Razzak, I., Daud, A., Aljohani, N. R., and Xu, G. (2019). "What's happening around the world? a survey and framework on event detection techniques on twitter". In: *Journal of Grid Computing* 17.2, pp. 279–312.

Shang D, X.-Y. D., Weiyi Ge, S. H., and Chen, J. (2018). "A Multi-view Clustering Model for Event Detection in Twitter". In: *Computational Linguistics and Intelligent Text Processing*, pp. 366–378.

Sheather, S. J. and Jones, M. C. (1991). "A reliable data-based bandwidth selection method for kernel density estimation". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 53.3, pp. 683–690.

Shiffler, R. E. (1988). "Maximum Z scores and outliers". In: *The American Statistician* 42.1, pp. 79–80.

Suma, S., Mehmood, R., and Albeshri, A. (2017). "Automatic event detection in smart cities using big data analytics". In: *International Conference on Smart Cities, Infrastructure, Technologies and Applications*. Springer, pp. 111–122.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). "Hierarchical dirichlet processes". In: *Journal of the american statistical association* 101.476, pp. 1566–1581.

Weng, J. and Lee, B.-S. (2011). "Event detection in twitter". In: *2011 International aaai conference on web and social media*.

Zhang, Z., He, Q., Gao, J., and Ni, M. (2018). "A deep learning approach for detecting traffic accidents from social media data". In: *Transportation research part C: emerging technologies* 86, pp. 580–596.