A Combined Classification-Clustering Framework for Identifying Disruptive Events

Nasser Alsaedi, Pete Burnap and Omer Rana
Cardiff School of Computer Science & Informatics, Cardiff University
{N.M.Alsaedi, P.Burnap, O.F.Rana}@cs.cardiff.ac.uk

Abstract

Twitter is a popular micro-blogging web application serving hundreds of millions of users. Users publish short messages to communicate with friends and families, express their opinions and broadcast news and information about a variety of topics all in real-time. User-generated content can be utilized as a rich source of real-world event identification as well as extract useful knowledge about disruptive events for a given region. In this paper, we propose a novel detection framework for identifying real-time events, including a main event and associated disruptive events, from Twitter data. The approach is based on five steps: data collection, pre-processing, classification, online clustering and summarization. We use a Naïve Bayes classification model and an Online Clustering method to validate our model on a major real-world event (Formula 1 Abu Dhabi Grand Prix 2013).

Keywords: Text Mining; Twitter Analysis; Machine Learning.

1. INTRODUCTION

In the recent years, Microblogging, as a form of social media, is fast emerging tool for expressing opinions, broadcasting news, and interaction between people. One of the most representative examples is Twitter, which allows users to publish short tweets (messages within a 140-character limit) about any subject. Real-life events are reported in Twitter too as users contribute content for a wide variety of events. The range of widely known events can be community-specific events, such as local gatherings, or can be wider-reaching national or even international level events. For example, the Iranian election protests in 2009 were extensively reported by Twitter users [1, 11]. Another good example, where Twitter was employed as a resource for the US government to communicate with citizens, was the swine flu outbreak when the US Centre for disease control (CDC) used Twitter to post latest updates on the pandemic [12].

Social media data present several challenges for event detection; the speed and volume at which data arrives, where tweets arrive continuously in a chronological order, and the size of the Twitter network produces a continuously changing, dynamic corpus. The significant amount of "noise" presented in the stream constitutes around 40% of all *tweets*, which have been reported as pointless "babbles" [3] like "let's go to the beach the weather is amazing". In fact, many posts do not provide any useful information or are spam where each post is short, which means that not much context is available for analysis. Moreover, space and time limitations arise from processing stream of documents at a very fast rate.

Nevertheless, Twitter has become a rich source of breaking news, including local news that are possibly of limited interest to wider global audience. When it comes to events, people tend to comment on real time events if a topic suddenly draw their attention (identified as spike or burst in activity), for example, sport events, weather, news, etc. Some topics are event-related, where as others are not related but they are popular (new released movie or album). Not only is Twitter significant because of its real-time characteristics, but also because it usually reports events ahead of newswire [4]. Therefore, several researchers have focused on identifying events in social media using different techniques [4, 9, 13-18, 22, 25, 29].

In this paper, we propose an online classification-clustering framework, which is able to handle a constant stream of new documents with a threshold parameter that can be modified in an experimental manner during training phase. The high volume of tweets from Twitter is the input of the system, which produces a table of the main events in a particular region, associated sub-events (details) and disruptive events for a particular time (daily or hourly manner). Social media data are very noisy; hence the first step in our framework after collecting data is preprocessing, which aims to reduce the amount of noise before classification. The next step is to separate event-related tweets and non-event content, here Naive Bayes Classifier is used as a classification method. Then, we compute messages' features in order to extract similar characteristics and apply incremental online clustering algorithm to assign each message in turn to a suitable event-based cluster after calculating tweet's similarity to the existing clusters, ultimately enabling us to detect disruptive events.

We focus in this work on online real-world events identification for both large scale and rare events such as car accidents in a given location, our contributions can be summarized as follows:

- Using our framework, we identify the relationship between social media activity and real-world events, and we detect the key events throughout the day. No prior knowledge is required about the number of events, their nature or popularity.
- Using our approach, we distinguish between the main event, the topic of the event, and sub-events we call *disruptive* events. Events are identified at a given place for a particular time.

 ISBN: 978-1-62561-000-3

- We validate our model on a major real-world event (Formula 1 Abu Dhabi Grand Prix 2013) to show the effectiveness of our framework. Our approach enables the identification of disruptive events with an average precision of 84% and of 80% over all other real-time events.

2. EVENT DETECTION

Identifying events from social media streams requires us to define an event. Wenwen-Dou in [15] provides a good definition of an event as "An occurrence causing change in the volume of text data that discusses the associated topic at a specific time". Here, we use the same definition where events have different degrees of importance causing the different "volume change" when discussed in social media platforms. Moreover, an event can be characterized by one or more of the following attributes: Topic, Time, People and Location [5, 17]. These attributes give details about an event and analyze the 4w questions: when, what, who and where [15].

One of the key questions in this paper is whether we can identify disruptive events from social media action such as protests, terrorist attacks, transport loss etc, as well as all the key moments and the development of sub events associated with it. So first we need to come up with a definition of a disruptive event on the context of social media as:

Disruptive event: an event that interferes the achieving of the objective of an event or interrupts ordinary event routine. It may occur over the course of one or several days, causing disorder, destabilizing securities and may results in a displacement or discontinuity.

For example, if a factory is likely to shut down due to a demonstration or by huge fire, related companies may get involved or even contact their customers in order to prevent unexpected losses or long delays. Therefore, monitoring meaningful patterns in social media and identifying abnormalities over time allows organizations or even governments to react to negative activities reported via online social networks such as Twitter to mitigate effects in a timely fashion before they escalate and potentially become damaging to wider society and business.

Experimentally, events can be characterized by burst detection or tweet/retweet ratio change where if passing a larger quantity of information, a link (URL) will be detected and possibly the inclusion of hashtags. However, detecting small scale rare events like car crashes, there are only small bits of information that surely includes additional challenges for discovering relevant information. Indeed, most disruptive events are inherently unpredictable events while, some of them events are controllable (traffic accidents) others are uncontrollable (natural disasters) [2, 7, 12]. Despite of all challenges, early detection of disruptive events is valuable for enrichment information intelligence and emergency management. Figure 1 compares between tweets ratio of a sport event (Sebastian Vettel victory in F1) and two disruptive events (traffic accidents and fire incidents) for the same period in the city of Abu Dhabi.

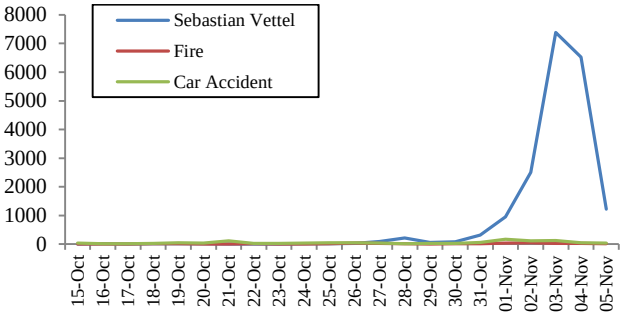

Fig. 1 Tweets volume per day mentioning "sport Event" "traffic accidents" and "fire incidents" in Abu Dhabi reported in Twitter

3. RELATED WORK

In the recent years, many researchers have shown interest in online event detection on social media. Many of the social media event detection were inspired by the previous work on event identification in textual traditional news (e.g. newswire). By using different methods for identifying social media content including machine learning algorithms, language models, feature-based algorithms and many more with distinctive goals to detect known events [11,12,13], unknown events [4,9,14,16,22] and even rare events [2,7,20,25].

Petrovic et al. [4] presented an approach to detect first story from a stream of tweets. The proposed approach, which is based on the locality-sensitive hashing (LSH), automatically organizes every incoming tweet in an existing story or labels it as a new story. In order to reduce the search space and improve the performance of the LSH, they added a secondary search which indeed improves the results by19%. However, this approach does not differentiate whether the new event is news, local event, natural disaster or just celebrity update.

Sakaki et al. [13] developed a probabilistic spatiotemporal model to monitor tweets and to detect disastrous events such as earthquakes. Their method is based on features such as the keywords "Earthquake!" or "Now it is shaking" where they assumed that each user is regarded as a sensor with a function of detecting target event and reports it in Twitter. One presumption of the approach is that users have to know the event in advance to provide representative keyword queries to be detected.

Becker et al. [22] proposed an online clustering framework, suitable for large-scale social media sites such as Twitter, to identify different types of real-world events and their associated social media documents. The online clustering technique groups together topically similar tweets and implements four features (Temporal Features, Social features, Topical Features and most importantly Twitter-Centric Features) to distinguish between real-world events and non-events. However, the framework is limited to widely discussed events and ignores rare events under predefined thresholds.

Recently, Burnap et al. [25] detected different levels of tension over time between online communities in Twitter using a Web Observatory platform (The Cardiff Online Social Media Observatory (COSMOS)). They implemented three

common approaches; text-based machine learning algorithms, lexicon-based methods and linguistic analysis and visualized tension levels as spikes over time. Furthermore, not all tweets are credible; Twitter also passes a negative by-product incorrect information as a large percentage can originate from spammers and people retweeting rumors [23].

In contrast to the aforementioned mentioned approaches, our goal is to automatically identify as many real-world events in a given region without any previous assumptions about events also our approach is not restricted to specific language. Our approach uses online clustering with sliding window timeframe which can be generalize to detect global and local events from social media streams with particular attention of disruptive events. Additionally, disruptive events are widely discussed in social media such as severe weather conditions (e.g. fog, storms) but sometimes there are only reported by few users such as car accidents and labor strikes.

4. FRAMEWORK FOR EVENT DETECTION

As we receive high volume of tweets per day with wide variety of tweets, traditional monitoring and analyzing is impractical as well as it significantly reduces the set of potentially applicable real-time algorithms. Identifying events and their associated documents over social media streams is a challenging task, yet information describing events from users can be critical in many situations and for purposes of gathering information about the ongoing events in a given area. Figure 2 shows the framework, which allows automatically identifying meaningful events from social media, preferably with a minimal number of non-important events. The method is based on collecting a series of data over timing frame windows for a given location. Five steps framework includes; data collection, pre-processing, classification, on-line clustering and summarization.

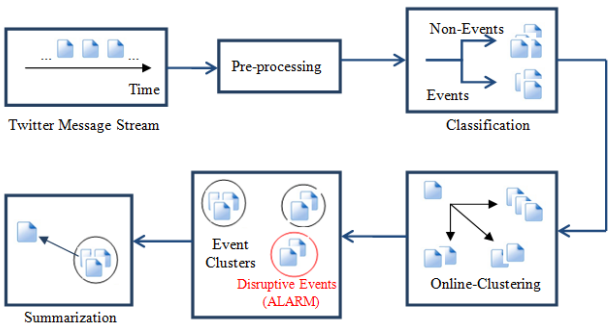

Fig. 2 Twitter Stream Event Detection Framework

4.1 DATA COLLECTION

In this study, our dataset contains collected tweets from 15/10/2013 to 5/11/2013 using Twitter streaming API as it allows subscribing continuous live stream of new data. Our initial aim was to monitor and analyze disruptive events associated with major occasions in a particular region. Hence, we have chosen the occasion to be (FORMULA 1 GRAND PRIX 2013) which was hosted in Abu Dhabi between (1-4/11/2013) but we extracted data for 15 days before the event to identify the differences in sports messages reported before the event and during the event in Twitter as well as to train the online clustering algorithm and to set the thresholds.

We collected tweets based on a set of keywords that describe Abu Dhabi and sport in general in different languages practically in Arabic and English. We also collected tweets from users who selectively add Abu Dhabi (or the surrounding cities in the UAE) as their location. Figure 3 shows the tweets volume in Abu Dhabi which clearly indicates the rise of sport posts during the F1 event. Figure 3 also shows an increase in the total frequency of all tweets in Abu Dhabi for F1 period because of its popularity and due to the various associated events such as financial events, entertaining events, disruptive events etc.

Data is stored using MongoDB [38], an open-source document database, easy to use and provides high availability speed and memory. In addition, MongoDB is suitable to store tweets, supports different indices with straightforward queries [38]. We store all collected tweets for 24 hours, similarly inactive clusters which are not updated within 24 hours are erased.

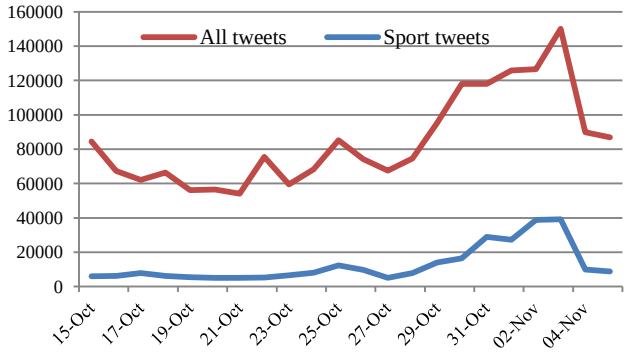

Fig. 3 The volume of tweets in the data set from (15th Oct to 5th Nov) in Abu Dhabi

4.2 PRE-PROCESSING

The goal of pre-processing of the collected data is to represent it in a form which can be analyzed efficiently and to improve the data quality by reducing the amount of noise (i.e. deleting tweets that are irrelevant to events).

We perform traditional text processing techniques such as stop-word elimination (Term frequency and TF-IDF are the criterions used for classifying stop words) and stemming (Khoja stemmer for Arabic tweets [26] and Porter Stemming [27] for English tweets). Moreover, posts which are less than 3 words are removed and tweets with one word accounted for over half of the words are also removed as these posts are less likely to contain useful information.

4.3 CLASSIFICATION

After pre-processing of the data, classification step aims to distinguish real-time events from noise or irrelevant tweets. Thus, the purpose of this step is to reduce the amount of noise from the incoming tweets and filter out as many non-event tweets as possible. Here, words of each tweet are considered

as features and a Naive Bayes Classifier similar to [16] was chosen over a number of other methods due to its performance in our experiments (results are shown in section 5.1).

The main reasons for using Naïve Bayes model are; regardless of its simplicity, it has been shown to be a very powerful model [9, 16, and 25]. Naïve Bayes model has many advantages such as it is relatively fast to compute, easy to construct with no need for any complex iterative parameter estimation schemes. Unlike SVMs or Logistic Regression, Naïve Bayes classifier treats each feature independently. Naïve Bayes also tends to do less overfitting compared to Logistic Regression [9]. However, the strong assumption of conditional independence between features reduces the power of Naive Bayes.

We used the R statistical software package[1], specifically the e1071 R package, to build and train the Naïve Bayes Classifier on a training corpus of 1500 tweets that have been annotated as "event" or "non-event". Given a tweet t represented as a set of words $w_1, w_2 \dots w_k$, the probability that t is an event is denoted by $\mathrm{P}(\mathrm{E}|\, w_1, w_2 \dots w_k)$, which can be rewritten as follows using Bayes' theorem:

$$\mathrm{P}(\mathrm{E}|\, w_1, w_2 \dots w_k) = \mathrm{P}(\mathrm{E}).\frac{\mathrm{P}(w_1, w_2 \dots w_k|\mathrm{E}\,)}{\mathrm{P}(w_1, w_2 \dots w_k)}$$

Similarly, given a tweet t, the probability that it is a non-event tweet is given by $\mathrm{P}(\mathrm{N}|\, w_1, w_2 \dots w_k)$, which can also be rewritten using Bayes' theorem:

$$\mathrm{P}(\mathrm{N}|\, w_1, w_2 \dots w_k) = \mathrm{P}(\mathrm{N}).\frac{\mathrm{P}(w_1, w_2 \dots w_k|\mathrm{N}\,)}{\mathrm{P}(w_1, w_2 \dots w_k)}$$

Using the assumption of independence among the words in t as well as our prior calculations of P(E), P(N), $\mathrm{P}(w_i|\mathrm{E}\,)$, and $\mathrm{P}(w_i|\mathrm{N}\,)$, we introduce the threshold (D) :

$$D = log\,\frac{\mathrm{P}(\mathrm{N}|\, w_1, w_2 \dots w_k)}{\mathrm{P}(\mathrm{E}|\, w_1, w_2 \dots w_k)} = \log(\frac{P(N)}{P(E)}) + \sum_{i}^{k} \log\frac{\mathrm{P}(w_i|\mathrm{N}\,)}{\mathrm{P}(w_i|\mathrm{E}\,)}$$

If D < 0, then the tweet is classified as event, else the tweet is classified as non-event and discarded.

4.4 CLUSTERING

After classification was performed, documents related to real-world events and non-real world events should be separated where non-events (such as chats, personal updates, incomprehensible messages, spam) are mostly filtered. Hence the input for the clustering stage is the output of the Naïve Bayes Classifier and includes only those tweets classified as being related to an event. To identify the topic of an event, while also determining those that are disruptive sub-events, we define a wide range of features including temporal features, spatial features and textual features, which are detailed in this section. We then apply an online clustering algorithm similar to [22, 26]. The decision to use an online clustering algorithm was taken for three key reasons; firstly, the online clustering algorithm supports high dimensional data as well as handles the large volume of data coming from social media. Secondly, many clustering algorithms such as K-means require the prior knowledge of the number of clusters whereas the online clustering approach does not require such knowledge. Finally, partitioning algorithms are ineffective in this case because of the high and constant sheer scale of tweets [22].

4.4.1 FEATURE SELECTION

Many researchers have proposed enhancements to models, computation improvements or develop new approaches to optimize the capturing of patterns in the input signals. Here, we compute many features related to the Twitter streams in order to reveal characteristics of clusters that are associated with real-world events.

Temporal feature

Temporal feature is an important factor that has been ignored by many studies not only in clustering but also in classification domain. Especially in social media where users and authorities are interested in the latest information hence a dynamic environment. Keeping an assumption in mind, some very quality tweets in the past may not be as important as in the present or in the future [19]. This is the reason behind keeping the most frequent terms in the cluster into hourly time frame window which characterize the frequent clusters. By comparing the number of messages posted during an hour which contain term t to the total number of messages posted during that hour. Not only temporal dimension enable events clustering but also it helps us to order events which is a challenging problem itself especially when having multiple events (One is dependent on the other event, or in case events have cause-effect relationship, or an event is longer than the other event). Figure 4 shows the temporal feature of "Sebastian Vettel" before and during his victory in 2013 FURMULA 1 Abu Dhabi.

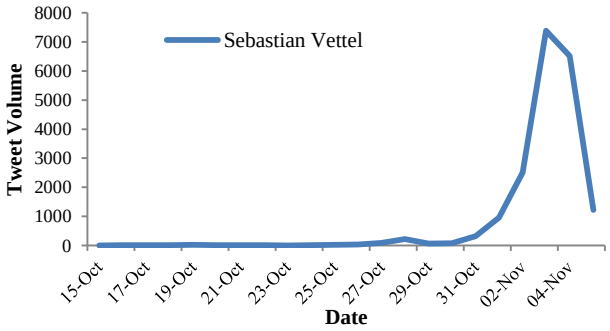

Fig. 4 Tweet volume associated with "Sebastian Vettel" from 15th Oct -5th Nov

Spatial feature

Events are usually characterized by rich set of spatial and demographic features [20]. Actually, the spatial dependency is important in early stage event detection [21]. In this paper, we make use of three techniques to extract geographic content from clusters. The first one is from Twitter where the source latitude and longitude coordinates are provided directly from

[1] http://www.R-project.org/

the user. The second method depends on the shared media (photos and videos) by using the GPS coordination of the capture device (if supported). The third method is to use the Named-Entity Recognition (NER) for geo-tagging the tweet content (text) which enhances the identification of places such as location, organization, street names, landmarks etc.

Once the geographic content has been extracted from each tweet in a cluster, we aggregate them to determine the cluster's overall geographic focus. The higher the volume of tweets from approximately near coordinates, the higher the level of confidence will be.

Textual features:

- **Near-Duplicate measure**

We compare the cosine similarity of tweets in each cluster; if two tweets have a very high similarity (0.95) we assume that one of them is a duplicate of the other. The original tweet is considered as the first tweet in a particular time frame and/or the shortest tweet in length. Even though duplicates are believed to be disadvantage (newer messages do not add any unique information), several users independently witnessing an event and tweeting about it, that would effectively increase the confidence level of an event.

- **Retweet ratio**

Cluster that contains a high percentage of retweets, especially from a single post by a celebrity, may not contain real-world event information [22]. But since most non-event tweets are assumed to be filtered out in the classification step, Retweet ratio can indicate events where users either agree with the message or wish to spread the information with more users. Indeed, Retweet ratio has been implemented to detect events and to estimate rumors in social media stream [23].

- **Mention ratio**

A mention is mechanism used in Twitter to reply to other users, engage others or join a conversation in a form of (@username). A user can mention one or more users anywhere in the body of the post. Hence, simply we calculate the number of mentions (@) relative to the number of tweets in a cluster.

- **Hashtag ratio**

Hashtags are important feature of social networking sites which can be inserted anywhere within a message: before, within or after the body of a message as a postscript. Some Hashtags indicate their posted messages (#bbcF1) and some others are dedicated originally to events such as (#abudhabigp). In addition, topic hashtags are used as search key on Twitter track interface to proactively search Twitter for more tweets belonging to a particular topic [16]. Indeed, the use of hashtags became the central coordinating mechanism for disaster-related user activity on Twitter [24].

- **Link or Url ratio**

Twitter is limited to 140 characters per message which add more importance to words in a tweet. In fact, it is common in twitter community to include links or shorten links when tweeting to refer to detailed information or to share additional knowledge. For tweets in a cluster having links to the same website may confirm that these tweets refer to the same topic. Therefore, the co-occurrence of URLs is especially significant in topic detection.

- **Semantic Category**

In the clustering step, there exist some of the famous event categories such as "politics", "sports" , ... which are more likely to occur most of the time. Semantic Category indicates whether the new cluster belongs to existing categories and merges them together. We use this feature to reduce the number of clusters in the algorithm.

- **Present Tense and Semantic nouns**

One of the main goals of this paper is the ability to detect messages that contain precise information about rare disruptive events such as labor strike or fire in a manufacture. To enrich such rare event identification, present tense and popular nouns that describe events as they take place should be taken as a feature. This is a dictionary-based feature that uses a selection of manually labeled dictionaries that were created by us.
Examples of present verbs are: witness, notice, observe, participate, engage, perform, listen etc.
Examples of Semantic nouns are; live, urgent, breaking news, latest, update etc.

4.4.2 ONLINE CLUSTERING ALGORITHM

The objective of online clustering is to automatically assign each document into a cluster according to textual similarity measures without a prior knowledge of the number of clusters or the nature of the real-world events. An event is a vector, where each dimension is the probability of feature in the event. Each tweet is represented as a TF-IDF weight vector of its textual content, and cosine similarity metric is used as the clustering similarity function *E*.

For a set of features $(F_1,\ldots,F_k)$ of the documents $(D_1,\ldots,D_n)$ and using their appropriate similarity measures different clustering solutions $(C_1,\ldots,C_k)$ can be formed using the following procedure:

- Given a threshold τ, a similarity function *E* and the data points to cluster $D_1,\ldots,D_n$, this algorithm considers each data point D_i in turn and computes its similarity $E(D_i, c_j)$ against each cluster c_j, for $j=1,\ldots,m$, where m is the number of clusters (initially $m=0$).
- If no cluster is found with the centroid whose similarity to D_i is greater than τ, then a new cluster is formed containing data point D_i and with the centroid value as the value of D_i.
- Otherwise, D_i is assigned to the cluster which gives maximum value for $E(D_i, c_j)$ and after adding D_i to cluster j new value of c_j is computed.

The centroid of a cluster which is the average weight of each term across all documents in the cluster is used in this paper. The threshold parameters are determined empirically in

the training phase, however human interaction can also be useful to alter the threshold manually if needed in order to detect particular events from the stream.

The feature vectors are calculated according to feature selection for the calculation to be feasible (i.e. the calculation is limited to 60 minutes time window and for a maximum of approximately 100 miles variance). For a set of known locations where the prime location is the city of Abu Dhabi in our case that is characterized by streets' names, organizations, popular buildings and geographical areas. These names and data are provided by Abu Dhabi Spatial Data Infrastructure (AD-SDI) [2] who are the specialists in Abu Dhabi GIS (Geographic Information System).

One of the questions that we address in this paper is: Can we identify disruptive events from the data stream? Some of disruptive events are widely discussed in the social media such as (severe weather and its influence on the transportation sector) whereas some others are rare and concern only a small group of users such as car accident that add extra challenges. Feature selection is used in our framework to enrich the identification of such events.

Additionally, we manually boost the system with collection of 315 keywords which we believe are of substantial importance to disruptive events in social media.

4.5 SUMMARIZATION

Summarization or in our case cluster representation is the last stage of our framework, which should produce some sort of summary of each cluster. Summarization task is very challenging task in its own and takes various forms such as event summarization, text summarization and micro-blog event summarization [35]. After an event has been detected and assigned to a cluster; our goal is to extract the most representative tweet from that cluster. The simplest approach to summarizing tweets is to consider each tweet as a document, and then apply a summarization method on this corpus to capture its key features [8, 15, 16, 35, and 36]. A more complicated approach is the one proposed by Chakrabarti and Punera where they use a variant of Hidden Markov Models to obtain an intermediate representation for a sequence of tweets relevant for an event [34].Another totally different approach is to implement Phrase Reinforcement Algorithm as proposed by Sharifi et al in [25] to find the best tweet that matches a given phrase, such as trending keywords. Voting algorithms [37] are utilized in many applications where in the context of social media can be considered taking into account the following features:

- The average length of a tweet.
- The total frequency of features in a tweet.
- Number of times of retweets, favorites and mansions of a tweet.
- Tweet that includes multimedia file such as photo, video or URLs.

In this paper, we implement a voting selection approach where the highest number of retweets is utilized as a measure of summarization task however we leave the improvement of social media summarization for future work.

5. EXPERIMENTS AND RESULTS

5.1 EXPERIMENT 1

The aim of this experiment is to elect the best classifier between different machine learning algorithms for the purpose of identifying events and non-events tweets. We have chosen three well-established machine learning algorithms; Naive Bayes classification a statistical classifier based on the Bayes' theorem (further details in section 4.3), Logistic Regression, a generalized linear model to apply regression to categorical variables [28] (details about Logistic Regression [29]), and support vector machines (SVMs) which aims at maximizing (maximum margin) the minimum distance between two classes of data using a hyperplane that separates them (for the full algorithm refer to [30]).

From our collected data, we manually labeled 1500 tweets in to two classes "Event" and "Non-Event" to train our classifiers. Event instances outnumber the non-event ones as the training set consisted of 600 Non-Event tweets and 900 event-related tweets. 200 of event-related tweets contain specific keywords for "disruptive event" category like severe weather, car crashes, protests, strikes, fire incidents ... to enhance the identification of disruptive events. In spite of the fact that misclassifying number of events to non-event could affect the accuracy of the classifier, it substantially improves the identification of real-world events. Agreement between our two annotators, measured using Cohen's kappa coefficient, was substantial (kappa = 0.825).

A ten-fold cross validation approach [25, 28] was used to train and test the machine learning methods. For each evaluation, the dataset is split into 10 equal partitions and trained 10 times. Every time the classifier is trained on 9 out of the 10 partitions and uses the tenth partition as test data. In addition, for the classification task, we have used the WEKA machine learning toolkit[3] because it contains a whole collection of machine learning algorithms for data mining tasks including testing, analyzing, comparison and the automatic calculation of performance measures.

Here we adopted a set of well-known performance measures for text classification: precision (how often are our predictions for a class are correct —a measure of false positives); recall (how often tweets are classified correctly as the correct class — a measure of false negatives); F-measure, a harmonic mean of precision and recall; and accuracy, the proportion of the correctly classified tweets to the total number of tweets which measure the overall effectiveness of a classifier. For a result set, we have:

[2] http://sdi.abudhabi.ae/

[3] http://www.cs.waikato.ac.nz/ml/weka/

tp(true positive)	fp(false positive)
fn(false negative)	tn(true negative)

$$\text{Precision(P)} = \frac{tp}{tp + fp} \qquad \text{Recall(R)} = \frac{tp}{tp + fn}$$

$$\text{F} - \text{measure} = \frac{2 \times P \times R}{P + R} \qquad \text{Accuracy} = \frac{tp + tn}{tp + fp + fn + tn}$$

Table 1 show a comparison of classifiers with unigram presence which clearly indicates that Naive Bayes classifier produces the best results.

Naive Bayes classifier		**Human**	
		Yes	No
	Yes	683	164
	No	104	549

SVM Classifier		**Human**	
		Yes	No
	Yes	701	177
	No	109	513

Logistic Regression classifier		**Human**	
		Yes	No
	Yes	646	234
	No	129	496

Table 1 Accuracy, Precision, recall and F-measure for different classification algorithms.

	Naive Bayes classifier	**SVMs classifier**	**Logistic Regression classifier**
Accuracy	82.13	80.93	76.13
Precision	80.64	79.84	73.91
Recall	86.79	86.54	83.90
F-measure	83.60	83.05	78.30

Furthermore, we aim to investigate methods to improve the performance of the classification results, thus we consider different features which capture patterns in the data such as n-gram presence or n-gram frequency, the use of unigrams, bigrams and trigrams, linguistic features such as parts-of-speech (POS) tagging and Named Entity Recognition (NER). Some researchers have reported that best performance is achieved using unigrams [31], while other works report that bi-grams and trigrams outperform unigrams [32]. However they are agreed that term-presence gives better results than term frequency for instance [33] shows that the presence of words only once in a given corpus is a good indicator of higher precision. In addition, the part-of-speech (POS) tagging, a basic form of syntactic analysis, used to disambiguate sense in many applications in natural language processing (NLP) while, Named Entity Recognition (NER) is used to extract proper names or entities from a given corpus such as persons, organizations, and locations. Here we used the Standford PoS tagger[4] because it has English tagger model, Arabic tagger model and other tagger models for several languages.

The classification accuracies' results from table 2 using bigram as features show that the performance of Naive Bayes and SVMs classifiers does not improve beyond that of unigram, but there is a noticeable improvement in the case of Logistic Regression.

[4] http://nlp.stanford.edu/software/tagger.shtml

Features	**Naive Bayes classifier**	**SVMs classifier**	**Logistic Regression classifier**
Unigrams	**82.13**	80.93	76.13
Bigrams	**79.52**	78.18	78.57
Trigrams	72.84	**74.09**	69.97
Unigrams + Bigrams	**83.67**	82.23	79.45
POS + NER	**83.50**	81.92	81.38
Unigrams + Bigrams+ POS + NER	**85.43**	83.86	80.22

Table 2 Comparison of classification accuracies of different classification algorithms over set of features.

In addition, the classification accuracies of all three classifiers have been declined when using trigrams as features which provide suggestive evidence that the use of n-grams for Twitter classification might not be a good approach due to the limitations on the size of tweets. Hence the elimination of the use trigram and higher order of n-gram and instead we combine unigrams and bigrams in order to improve performance by getting the best of unigrams and bigrams. Indeed, Naive Bayes classifier achieved an accuracy of 83.67% as well as we got a boost of approximately 1.3% in SVMs and an improvement of about 3.3% in the case of using Logistic Regression classifier.

The use of both part-of-speech (POS) tagging and Named Entity Recognition (NER) have resulted in better performances as they help in a better understanding of how words are related to events and they also differentiate between different senses of a word (word-sense disambiguation). The final test combines all the successful features (Unigrams + Bigrams+ POS + NER) which lead to the highest classification accuracy achieved by Naive Bayes classifier of 85.43%.

5.2 EXPERIMENT 2

The resulting dataset after classification contains around 85,000 event-related tweets which we used to train, test and evaluate the clustering algorithm. We used the first 15 days of data (from 15/Oct until 29/Oct) to train the clustering algorithm and to tune the thresholds using the validation set. Then we tested the clustering algorithm on unseen data of the last 6 days from the 30th of Oct until the 4th of Nov. In this experiment, we have used all features (from section 5.4.1) where the best selection of features is reserved for future work. Not all features are expected to improve system's performance or lead to more accurate discrimination of the clustering algorithm. In fact, including some features could result in worse system's behavior then they should be removed. Moreover, we noticed that training algorithm with multiple features can result in some scalability issues. Table 3 summarizes results achieved using our framework on the test set by showing the number of events related to known category divided into training set and test set.

Date	Politics	Finance	Sports	Entertainment	Technology	Culture	Disruption Events
30-Oct	29	10	16	10	7	2	9
31-Oct	23	6	22	13	3	4	5
1-Nov	22	9	18	25	6	12	12
2-Nov	18	8	20	26	9	5	9
3-Nov	17	7	20	18	5	7	7
4-Nov	13	9	10	11	7	6	3

Table 3 Number of real-world events obtained using the clustering algorithm on the test set

In order to evaluate the clustering performance, we employed two human annotators to manually label 800 clusters. The task of the annotators was to choose one category from eight different categories: politics, finance, sport, entertainment, technology, culture, disruptive event and other-event. The other-event category represents all other events which are not related to the above categories. We divided the test set into six datasets according to each day for annotation task. Annotators' task was to manually label clusters (not tweets) to obtain the total number of events per category per day.

The agreement between annotators was calculated using Cohen's kappa (K=0.794) which indicates an acceptable level of agreement. We used 635 clusters on which both annotators agreed as the gold standard. Therefore, evaluation is performed by computing average precision (AP) on the gold standard. Averaged precision measures (how many of the identified clusters are correct averaged over hours per day and calculated based on the precision of each cluster per day. Average precision is a common evaluation metric in tasks like ad-hoc retrieval [4, 10, 22, and 33] where only the set of returned documents and their relevance judgments are available. Table 4 shows the average precision percentages of the cluster on the test set.

Date	Politics	Finance	Sport	Entertainment	Technology	Culture	Disruption Events	Average Per Day
30-Oct	82.50	81.11	85.71	76.00	78.80	74.29	87.50	80.84
31-Oct	78.71	85.67	80.62	76.87	74.21	83.36	82.00	80.21
1-Nov	84.15	82.52	80.90	74.45	75.75	81.61	84.67	80.58
2-Nov	77.01	79.40	77.29	72.51	72.19	67.50	90.00	76.56
3-Nov	79.91	83.49	90.21	68.96	82.35	83.36	78.17	80.92
4-Nov	84.34	81.33	82.04	74.01	83.99	79.03	82.76	81.07
Average Per Topic	81.10	82.25	82.79	73.80	77.88	78.19	84.18	80.03

Table 4 Average precision of the online clustering algorithm, in percent.

In general, the online clustering algorithm was able to achieve a good performance; although, the performance was inconsistent with respect to topics. For example, the average accuracy of identifying sport events was greater than the average accuracy of identifying entertainment events by about 9%. In fact, it is easier to extract and categorize events like politics, finance, sport and disruptive events than events like entertainment, technology or cultural events even for humans which cause the main disagreement between annotators in the annotation task. The best performance achieved by the online clustering algorithm was in the case of the disruptive event identification of 84.18%.

We wished to compare our results with other works in the area of event detection on Twitter, but that is not possible due to the differences between datasets as each dataset has different size, time and characteristics. Furthermore, validating our results against real-time official reports or from news stream is not feasible at this point as we need to create a dataset of events from traditional media combined with officials reports about for instance disruptive events. Even if we attempt to create such dataset, the performance of our model will be lower for many reasons; firstly, not all events reported in traditional platforms are reported in social media and vice versa. Secondly, Twitter streaming API only allows 1% of the total number of tweets for researchers which mean that we fail to report the 99% of online conversations. Conversely, 1% is in fact a huge corpus of tweets per day for sampling and researching purposes. Lastly, we undoubtedly accept the limitations of our framework as it is capable of capturing events (like disruptive events) with few posts but cannot identify events with too few messages.

6. CASE STUDY

One of the framework's objectives is to identify disruptive events and send a notification to the administrators or users depending on the given permissions. Table 5 shows the top 3 emerging disruptive events identified by the framework based on the number of retweet counts for the F1 ABU DHABI dataset. For space limitation, we only present results of the disruptive incidents associated with the (3 days) of the actual race as an example of the system's output. Events and topics detected from social stream are different from what were covered on the same days in the traditional media, like news stream. Most of the disruptive events identified by the system were car accidents, fire incidents, weather warnings, labor strikes and rumor corrections. Furthermore, we believe that our techniques can support and enhance the decision making process using different types of user-generated content such as information gathering and small-scale incidents detection. Figure 5 illustrates the idea of detecting disruptive events by showing the number of tweets for two target events: "Road accidents" and "fire incidents" over time.

Date	Tweet	Translation	RT count	Comments
Nov 1	الان حادث على شارع الاتحاد في دبي والزحمه وصلت إلى جسر القرهود باتجاه الشارقة يرجي الخذ الحيطة والحذر#قروب العواصف pic.twitter.com/5fL367qzFF	Now an accident on Union Street in Dubai and the crowds arrived at Garhoud Bridge towards Sharjah, please take extra caution #group storms pic.twitter.com/5fL367qzFF	75	
	حريق ضخم في محطة لتوزيع الكهرباء في ابوظبي بالقرب من مصفح الصناعيه ونسال الله السلامة للجميع pic.twitter.com/kLLc4L0hoJ	A huge fire in an electricity distribution station in Abu Dhabi near musaffah industrial area we ask God for everyone's safety pic.twitter.com/kLLc4L0hoJ	49	
	Thewind is so strong that the waves are breaking over the shoreway o-o		22	
Nov 2	Warning of thick **fog** on #**AbuDhabi**-Al Ain road http://bit.ly/17n0ivdL #UAE		92	
	ازدحام غير طبيعي على المدخل الشمالي غربي من الحلبة يا جماعة وين الامن وين السلطات؟؟؟؟؟؟ ولا مدخل الفندق ما تقدر تدخل ولا تطلع زحمة زحمة	Abnormal congestion on the north-west of the circuit entrances where is the security where are authorities?????? nor the entrance to the hotel is estimated interference nor looked Traffic Traffic	34	
	قام مئات العاملين في شركة " " القابضة، العاملة في مجال الاستثمار بقطاع الإنشاءات والمقاولات، بالإضراب عن العمل يوم أمس الأحد لدعم مطالب بزيادة الرواتب #دبي #ابوظبي #الامارات	Hundreds of workers in the company, " " Holding, operating in the field of investment sector, construction and contracting, to go on strike on Sunday to support the salary increase demands #Dubai #Abu Dhabi, #UAE	9	The name of the company has been removed
Nov 3	A major fire broke out in maintenance area near the south zone in the early hours today; no casualties reported :(#F1 #AbuDabi			corrected by the officials after 2 hours
	11:42PM. #Traffic congestion& delays on Sheikh Zayed Tunnel for Motorists coming from Al Corniche outbound #AbuDhabi		32	Post by Abu Dhabi police using their official twitter account
	كل يوم حفلة كل يوم مجون وسهر وكل هذا في بلدنا المسلم؟!!! ياخي ما فهمت شو دخل الحفلات فالرياضة): لو في لاس فيقاس ما شفنا كل هالمصخرة تبا احتلال مو سياحة #ابوظبي #ياسلام	Every day party every day soiree and all this shamelessness in our muslim country?!!! I don't get it what is the relationship between concerts and sport :(if we are in Las Vegas, I doubt we would see the same sh** f***seems invasion not tourism #abudhabi #yaslam	14	

Table 5 Top 3 emerging disruptive events identified by the system according to the number of retweet for the F1 ABU DHABI from the 1st to the 3rd of Nov 2013

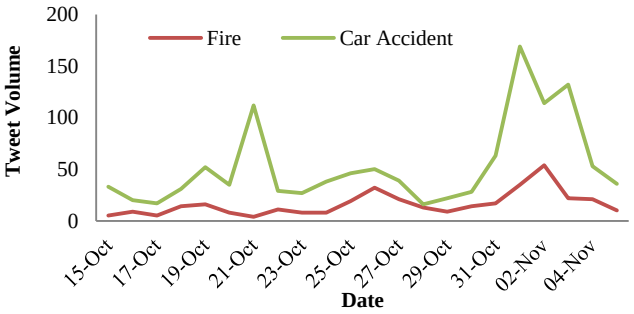

Fig. 5 Number of tweets reporting "road accidents" and "fire incidents" between 30/Oct to 4/Nov in Abu Dhabi

7. CONCLUSION

In this paper we have presented an integrated framework to detect real-world events on social media platform (Twitter). The event identification was performed through several stages; data collection, preprocessing, classification, clustering and summarization. We have also shown how our approach is able to reveal daily disruptive events for a certain location. Moreover, we have presented set of experiments and a case study to show the effectiveness of the proposed approach.

This framework can be generalized to develop a social awareness system or for the purposes of decision making enrichment which can be implemented in many fields such as crises management or information intelligence. Our results support the claim that the use of social media for the purposes of information gathering could be utilized as a complementary to traditional intelligence and not to be used independently. We accept the limitations of our system where improvements will be suggested and explored in the near future.

There are many directions for future work. One of the main directions is to compare and validate the performance of the proposed framework against other well known algorithms such as the state-of-the-art Labeled Dirichlet Allocation (LDA) method. Another direction is to investigate the contribution and the limitations of the various feature types to event detection in social media. Finally, the detection of rumors in social media, the analysis of the distinctive characteristics of rumors and the way they propagate in the microblogging communities will be carried out in the near future.

8. REFERENCES

[1] Kavanaugh, A., Fox, E. and Sheetz, S. Social media use by government: from the routine to the critical. *Government Information Quarterly* 29(4), pp. 480–49, 2012.

[2] Wang, X., Gerber, M. and Brown, D. Automatic crime prediction using events extracted from twitter posts. *Social Computing, Behavioral-Cultural Modeling and Prediction*, pp. 231–238, 2012.

[3] PearAnalytics. Twitter study. http://www.pearanalytics.com/wpcontent/uploads/2009/08/Twitter-Study-August-2009.pdf. August 2009.

[4] Petrović, S., Osborne, M. and Lavrenko, V. Streaming first story detection with application to twitter. *Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 181–189, 2010.

[5] Gundecha, P. and Liu, H. Mining Social Media: A Brief Introduction. *Tutorials in Operations Research, INFORMS 2012* (Dmml), pp. 1–17. 2012.

[6] Bruns, A. How long is a tweet? Mapping dynamic conversation networks on Twitter using Gawk and Gephi. *Information, Communication & Society*, pp. 15, 2012.

[7] Walther, M. and Kaisser, M. Geo-spatial Event Detection in the Twitter Stream. *In Proceedings of the 34th European Conference on Information Retrieval, ECIR 2013* 7814, pp. 356–367, 2013.

[8] A. Pak, P. Paroubek, Twitter as a corpus for sentiment analysis and opinion mining, *Seventh Conference on International Language Resources and Evaluation*, 2010.

[9] Khilnani, D., Khaitan, P. and Jin, Y. A Novel Approach to Event Duration Prediction. *nlp.stanford.edu*.

[10] Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.-H., Steinbach, M., Hand, D.J. and Steinberg, D. Top 10 algorithms in data mining. 2007.

[11] Zhou, Z., Bandari, R. and Kong, J. Information resonance on Twitter: watching Iran. *1st Workshop on Social Media Analytics (SOMA '10)*. 2010.

[12] J. Ritterman, M. Osborne, and E. Klein. Using prediction markets and twitter to predict a swine flu pandemic. *1st international workshop on mining social media*, 2009.

[13] Sakaki, T., Okazaki, M. and Matsuo, Y. Earthquake Shakes Twitter Users : Real-time Event Detection by Social Sensors. *19th International World Wide Web Conference (WWW '10)*. 2010.

[14] Cataldi, M., Caro, L. Di and Schifanella, C. Emerging topic detection on Twitter based on temporal and social terms evaluation. *Tenth International Workshop on Multimedia Data Mining*, pp. 1–10, 2010.

[15] Dou, W., Wang, X., Skau, D., Ribarsky, W. and Zhou, M.X. LeadLine: Interactive visual analysis of text data through event identification and exploration. *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 93–102, 2012.

[16] Sankaranarayanan, J. and Samet, H. Twitterstand: news in tweets. *Proc. The 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 42–51, 2009.

[17] Liu, X., Troncy, R. and Huet, B. Using social media to identify events. *The 3rd Workshop on Social Media (WSM'11)*, pp. 0–5, 2011.

[18] Albakour, M., Macdonald, C. and Ounis, I. Identifying local events by using microblogs as social sensors. *The 10th International Conference in the Open research Areas in Information Retrieval (RIAO)*, pp. 22–24, 2013.

[19] Yu, P., Li, X. and Liu, B. Adding the temporal dimension to search-a case study in publication search. *In Proceedings of Web Intelligence (WI'05). The 2005 IEEE/WIC/ACM International Conference*, p. 543,549. 2005.

[20] Wang, X., Gerber, M. and Brown, D. Automatic crime prediction using events extracted from twitter posts. *Social Computing, Behavioral-Cultural Modeling and Prediction*, pp. 231–238, 2012.

[21] Li, J. and Cardie, C. Early Stage Influenza Detection from Twitter. *arXiv preprint arXiv:1309.7340*, 2013.

[22] Becker, H., Naaman, M. and Gravano, L. Beyond Trending Topics: Real-World Event Identification on Twitter. *ICWSM*, pp. 1–17, 2011.

[23] Takahashi, T. and Igata, N. Rumor detection on twitter. *The 6th International Conference on Soft Computing and Intelligent Systems*, pp. 452–457, 2012.

[24] Bruns, A., Burgess, J., Crawford, K. and Shaw, F. # qldfloods and@ QPSMedia: Crisis communication on Twitter in the 2011 south east Queensland floods. (Cci). 2012.

[25] Burnap, P., Rana, O.F., Avis, N., Williams, M., Housley, W., Edwards, A., Morgan, J. and Sloan, L. Detecting tension in online communities with computational Twitter analysis. *Technological Forecasting and Social Change*. 2013.

[26] Khoja, S., Garside, R. and Knowles, G. Stemming Arabic text. *Proceedings of the Student Workshop at the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL2001)*. 2001.

[27] Porter, M. An algorithm for suffix stripping.pdf. *Program: electronic library and information systems* 40(3), pp. 211 – 218. 1980.

[28] Martinez-Romo, J. and Araujo, L. Detecting malicious tweets in trending topics using a statistical analysis of language. *Expert Systems with Applications* 40(8), pp. 2992–3000, 2013.

[29] Peduzzi, P., Concato, J., Kemper, E., Holford, T.R. and Feinstein, a R. A simulation study of the number of events per variable in logistic regression analysis. *Journal of clinical epidemiology* 49(12), pp. 1373–9, 1996.

[30] Joachims, T. and Dortmund, U. Making Large-Scale SVM Learning Practical. *In Bernhard Sch¨olkopf and Alexander Smola, editors, Advances in Kernel Methods - Support Vector Learning*, pp. 44–56. 1998.

[31] Pang, B., Lee, L. and Vaithyanathan, S. Thumbs up?: sentiment classification using machine learning techniques. *Empirical Methods in Natural Language Processing, EMNLP'02*. 2002.

[32] Dave, K., Lawrence, S. and Pennock, D. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. *Proc. of the 12th International conference on WorldWideWeb, ACM*. 2003.

[33] Yang, H., Callan, J. and Si, L. Knowledge Transfer and Opinion Detection in the TREC 2006 Blog Track. *The 15th Text Retrieval Conference (TREC 2006)* 120. 2006.

[34] Chakrabarti, D. and Punera, K. Event Summarization Using Tweets. (*ICWSM-2011)*. 2011.

[35] Chua, F. and Asur, S. Automatic Summarization of Events from Social Media. *(ICWSM-2013)*. 2012.

[36] Baeza-Yates, R. and Ribeiro-Neto, B. Modern Information Retrieval. *ACM Press/Addison-Wesley* 9, 1999.

[37] Bauer, E. and Kohavi, R. Empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning 38*, 1999.

[38] Kumar, S., Morstatter, F. and Liu, H. *Twitter Data Analytics*. Springer. 2014.