Huerta, Schade, Granell (Eds): Connecting a Digital Europe through Location and Place. Proceedings of the AGILE'2014 International Conference on Geographic Information Science, Castellón, June, 3-6, 2014. ISBN: 978-90-816960-4-3

Is this Twitter Event a Disaster?

André Dittrich
Institute of Photogrammetry and Remote Sensing, KIT
Karlsruhe, Germany
andre.dittrich@kit.edu

Christian Lucas
Institute of Photogrammetry and Remote Sensing, KIT
Karlsruhe, Germany
christian.lucas@kit,edu

Abstract

Social media services such as Twitter have become an important channel for reporting real-world events. For example, they can describe the current situation during a disaster. The decisions in crises management are based on detailed on-site information such as what is happening, where and when an event is happening, and who is involved. Thus, in real applications, monitoring the events over social media will enable to analyse the current overall situation. In this paper, the authors introduce a prototype for real-time Twitter-based natural disaster detection and monitoring. The detection approach is multilingual and calculates a statistical based probability for a potential disaster event. For an automatic geo-referencing of the disaster, the approach applies spatial gridding. On this basis the grid cells are subject to a spatial-thematic clustering which uses a method similar to region growing. The application's output is an automatically generated email alert, containing specific information on the disaster.

Keywords: social media monitoring, event detection, real-time analysis, disaster taxonomy, multilingual keyword search

1 Introduction

Online social media services, as Twitter, Facebook or Flickr, have changed the way of communication within communities and groups or between individuals. Monitoring and analysing this continuous flow of user-generated content can yield valuable information, which is not available from traditional sources. Twitters short messages (tweets) can be seen as a dynamic source of information enabling individuals, corporations and government organizations to stay informed. For instance, people are interested in getting advice, opinions, facts, or updates on news or events. Consequently, tweets can give the information to answer the usual 4W questions in the disaster management domain. *What* is happening, *where* and *when* an event is happening and *who* is involved. Within the 140 characters of a tweet the question about the *what* can be answered. The remaining information about the *where, when* and *who* need to be extracted from the tweet's metadata. The sender gives the *who-information*, the time stamp gives the *when-information* and the geo-reference gives the *where-information*.

This paper presents a prototype, which will monitor the Twitter stream and detect and analyse diverse kinds of natural disaster events.

The Twitter stream, however, also contains large amounts of meaningless messages, polluted content, spelling or grammatical errors, improper sentence structures and mixed languages, which negatively affect the detection process. To handle these effects a considerable amount of literature has been published on detection approaches. The authors classified the representative techniques for a short review in three categories according to the *event type*, the *detection task* and the *detection method*.

Depending on the event type, the techniques are classified into unspecified and specified event detection. Unspecified events of interest are typically driven by topics, that attract the attention of a large number of users. Because no event information is available numerous features that occur frequently are typically used to detect unknown events (cf. [9, 10]). In contrast, specified event detection aims on known or planned event types. These events could be specified by the related information such as location, time, or performers. The techniques attempt to exploit Twitter's textual content using a wide range of machine learning, data mining, and text analysis techniques (cf. [4, 8]).

According to the detection task, the techniques are classified into new event detection (NED) and retrospective event detection (RED) techniques. Most research is focused on NED, which involve continuous monitoring of signals to exploit the timely information provided by Twitter streams. Knowledge about the event is integrated into the detection, by using filtering techniques as [8] or using additional features such as the location (cf. [5]). RED techniques are more focused on chronological data. The search capabilities allow retrieving individual tweets in response to a query. Because relevant messages may not contain any query term and new shortcuts as hash tags may merge over time, the challenge is identifying relevant messages. So, event retrieval from Twitter data is often focused on temporal and dynamic query expansion techniques (cf. [6]).

Event detection from Twitter draws on different detection methods, including machine learning, data mining, natural language processing, information extraction and information retrieval. The major directions of the approaches are subdivided into *supervised*, *unsupervised* and *hybrid* approaches. Several supervised classification algorithms have been proposed for specified events, including for instance naive Bayes [9], support vector machines [8] or gradient boosted decision trees [7]. Most techniques for unspecified event detection from Twitter streams rely on clustering approaches as expectation–maximization algorithm [1] or threshold-based approaches as [9]. Above that, there are hybrid detection approaches proposed to identify Twitter messages. In [2], a factor graph model is used, which simultaneously extracts attributes of the event using a supervised conditional random field classifier. The review showed that the detection approaches are very specific regarding their respective aims. Thus they cannot directly be applied in a generic manner.

The detection technique of this paper is a training-based, statistical robust NED algorithm for a specific domain of events. In contrast to the introduced approaches, the prototype described here includes spatial-thematic clustering, temporal monitoring and classifies event types in multiple languages to meet the requirements of the disaster domain (cf. Section 3). The experimental results (cf. Section 4) show the system's performance based on various real-world events.

2 General Framework

This section will detail the data resource Twitter and the programmatically important characteristics of its API (Application Programming Interface). Additionally, the areas of investigation which are monitored by the system will be described

2.1 Data ressource

The developed application exploits Twitter as extensive data resource. Twitter provides real-time access to its worldwide ongoing traffic, called *Firehose*, through its Streaming API. However, only about 1% of all current tweets can be crawled for free.

Since a main focus of the analysis of an event is its location, the application only uses geo-referenced tweets. To send a geo-referenced tweet, the user needs to explicitly allow the Twitter client on his device to access the device's locational sensor (e.g. GNSS sensor). This results in approximately only 2% of all tweets worldwide being geo-referenced.

Besides the capability for keyword filtering, the API also allows for geospatial filtering in terms of bounding boxes. However, these methods are not applicable simultaneously. Nevertheless, these bounding boxes facilitate avoiding the 1% limit.

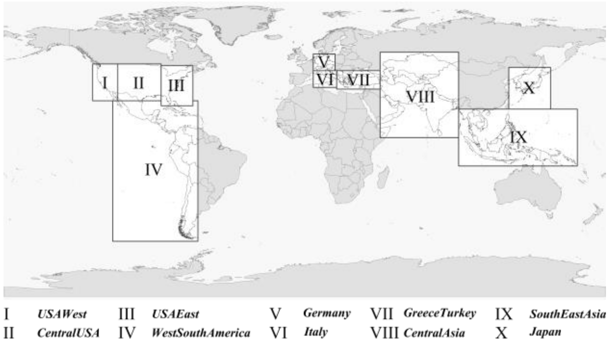
2.2 Investigation Area

The system is able to monitor any area around the globe for potential natural disasters, given a certain training period (c.f, Section 3.3). The areas are limited by a bounding box, due to Twitter's Streaming API constraints (cf. Section 2.1). Figure 1 shows the boundaries of the monitored test areas on a world map.

The areas were selected based on their potential risk of natural disasters such as earthquakes, volcanic eruptions, tornados, etc., and the popularity of Twitter in the countries they contain or overlap. For example, the main part of China is not included, as the Twitter service is blocked there by the Chinese government since 2009.

For example, the bounding box *WestSouthAmerica* (cf. Figure 1) was chosen because of its high risk of volcanic eruptions. The geographical extent ranges from 57° south to 27° north and from 115° west to 64° west.

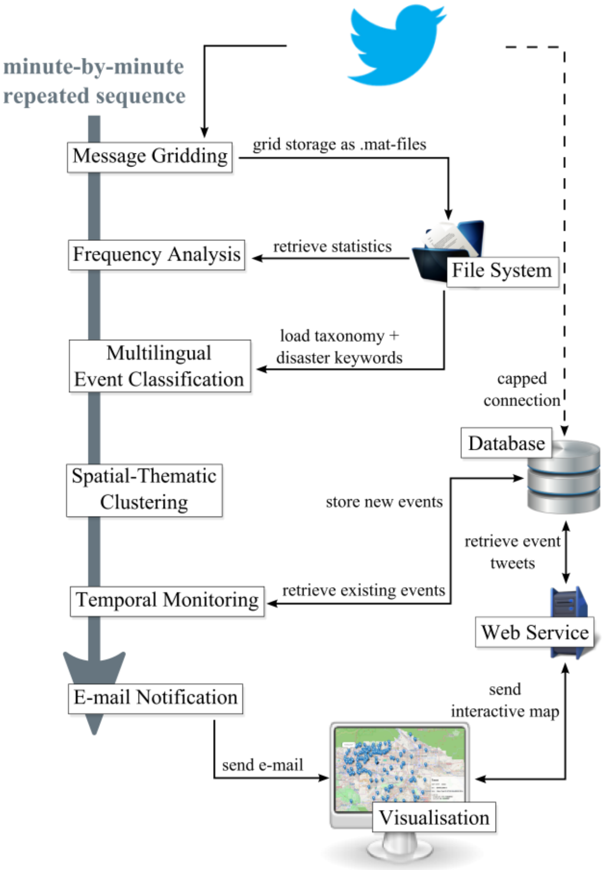
Figure 1: Map of the investigation areas and their names in the system



3 Analysis Workflow

This section will describe the implemented prototype with its analysis workflow in detail. The prototype is an automatic system for multilingual, real-time detection, classification, spatial-thematic clustering and temporal monitoring of natural disaster events. It is not a RED implementation, but a successfully running system that operates twenty-four-seven.

Figure 2: Prototype architecture and analysis workflow



3.1 Prototype Architecture

The implementation is primarily based on the Java programming language with embedded calls to Matlab scripts which e.g. execute numerical calculations. However, the main benefit of using Matlab is the efficient storage of sparse matrices as .mat-files (cf. Section 3.2). The visualization component is based on JavaScript.

Figure 2 depicts the complete architecture and analysis workflow of the implemented prototype exemplified for a single area of investigation. The basic requirement is a broadband internet connection to assure the access to Twitter's *Firehose*. However, the system is tolerant of communication disruptions to the Twitter service, as it immediately tries to re-establish the connection, usually successfully within a few seconds. Thus, even general network failures only result in the system not being able to detect events during that period. Shortly after the internet connection is restored the system will continue unaffectedly.

MongoDB a document-oriented, open-source database is used as data storage technology. As many other NoSQL technologies, it has the advantages of a dynamic schema. MongoDB uses the BSON (binary JSON) format, which is an enhanced version of the JSON format used by Twitter to provide their tweets via the API. Thus, the incoming tweets are stored *on-the-fly* without any further processing needed. Furthermore, the spatial indexing capability of the database allows for extreme fast retrieval of stored tweets based on their location data, i.e. their geographical coordinates.

The incoming tweets are stored in a collection with a capped connection, i.e. after a certain time frame (here 10 minutes) a tweet is deleted from the database. In contrast, the analysed events and their corresponding tweets are stored persistently.

3.2 Message Gridding

The foundation for the powerful detection mechanism is the mapping of the messages to a regular (numerically) 2-dimensional grid based on their geographical coordinates. Thus, also small-scale or regional natural disasters can efficiently be detected and do not disappear in the noise of the tweet baseline of the complete bounding box.

The chosen spacing of the grid points of 0.25° in the test areas is a balance between the speed of detection and the spatial granularity. Depending on the geographical latitude the spacing corresponds to an approximate distance of 25 to 28 km. Consequently, the cells of the grid are 0.25° × 0.25° and exactly cover the area of the respective bounding box. The incoming messages are assigned to the cell that covers the location where they were sent from. The temporal resolution of the system for the test areas is set to one minute, i.e. during one minute the messages are counted per cell and at the end of each minute the grid is stored as .mat-file with an according timestamp (date and time). In general, this time range ensures a sufficient number of tweets w.r.t. the chosen grid spacing to enable a robust statistical evaluation.

3.3 Training Phase

For a statistically robust and reliable detection, a training phase was conducted for each of the ten areas of investigation. In [3] is shown that the usual baseline of tweets of a specific region significantly differs between at least two types of days. This difference is strongly correlated with the percentage of Twitter users in this region who have to work on the next day. The highest accordance across all days of the week lies between 4pm and 6pm local time. To Account for these findings, a 24 hour period (i.e. a day) starts at 5 pm local time respectively. Moreover, the prototype distinguishes between 24 hour periods starting on a Friday, on a Saturday or on another day of the week. With the temporal resolution of one minute, the system stores 1440 grids per day and bounding box.

The training comprised 30 complete 24 hour periods for each of the three types of days. Thus, the mean value and the standard deviation of the amount of tweets could be derived for each cell for every minute of a day (and of course for each bounding box).

3.4 Frequency Analysis

The first indicator of an event in general, is an exceptional increase or decrease of the volume of tweets in a specific region. So far, the prototype only handles the case of increasing Twitter traffic, as it facilitates a meaningful and robust content analysis.

To decide whether an unusual high amount of tweets occurred in a cell during the time step of one minute, the counted number of tweets x is subject to a hypothesis test. Herein, x is checked against the mean value m and the standard deviation s of the cell in the preceding minute. The derived statistical values (m and s) from the training phase are automatically weekly updated in the running system. The null hypothesis H_0 in the test is defined as *no event happened*. The alternative hypothesis H_A consequently is defined as *an event happened*.

The significance level is set to $\alpha = 5\%$ and the power of the test is $1 - \beta = 90\%$. The test value is calculated as

$$k = \frac{x - m}{s} \sim N(0,1) \tag{1}$$

From the critical value for H_0

$$c_{H_0} = N(0,1)_{1-\alpha/2} \tag{2}$$

and H_A

$$c_{H_A} = N(0,1)_{1-\beta} \tag{3}$$

the decentralization parameter

$$\delta = c_{H_0} + c_{H_A} \tag{4}$$

is derived. Finally, if $k > \delta$, the systems accepts the alternative hypothesis H_A and thus identifies a significant rise in tweets in the respective cell.

3.5 Multilingual Event Classification

The next step analyses the content of the tweets in the cells, which were identified through the hypothesis test, to try to assign them a certain class or type of natural disasters.

Therefore, the system retrieves the tweets from the preceding minute and the respective cell from the database based on their timestamp and geographical coordinates. After that, the textual content of the tweets is scanned by a multilingual keyword search for terms related to natural disasters. The 133 terms, which are based on past event experience, were compiled in English. General disaster related terms are also included at this stage to provide a more comprehensive situational awareness for disaster managers.

For each of the 43 languages possibly occurring in the investigation areas (cf. Table 1), the complete list was translated with the aid of the MyMemory REST API[1] and Google Translate[2]. To assure real-time performance, each bounding box was assigned only the languages that are common in its geographic region plus English. For example, the languages for the *WestSouthAmerica* bounding box are Spanish, Portuguese and English, i.e. the German word *Erdbeben* (Eng. earthquake), would not be detected in this bounding box.

Table 1: List of the 43 languages in which the system can identify terms related to natural disasters

Arabian	Spanish	Japanese	Dutch	Tagalog
Azerbaijanian	Persian	Georgian	Polish	Telugu
Bulgarian	French	Khmer	Portuguese	Thai
Bengal	Hindi	Korean	Romanian	Tamil
Bosnian	Croatian	Laotian	Russian	Turkish
Cebuano	Hungarian	Macedonian	Slovak	Urdu
Czech	Armenian	Marathi	Slovenian	Vietnamese
German	Indonesian	Malaysian	Albanian	
Greek	Italian	Maltese	Serbian	

The occurrences of the identified keywords in the retrieved cell are translated back to English and added up term-wise. In result, this yields a list of disaster-related English terms with their respective absolute frequency (cf. Table 2).

For the classification of the event, a hierarchical tree structure (taxonomy) of natural disaster types was established. In this structure, the leaves represent disaster types, which are usually not further distinguished by non-experts in natural language. Each of these leaves is assigned a bag-of-words (BoW) that is virtually unambiguous for the specific event type (cf. Figure 3). Similar to the 133 general disaster terms, the BoWs are derived from investigations of tweets from past events. The union of all BoWs of the child nodes represent the BoW for the respective parent node, e.g. the BoW for the type *Hydrological* is the union of the BoWs of *Flood* and *Tsunami*.

Table 2: Example result of a cell for the multilingual keyword search

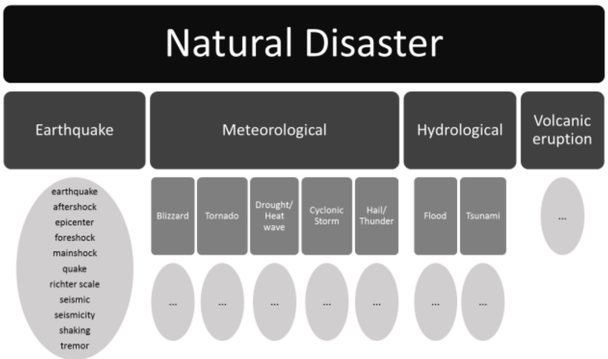
term	count
earthquake	10
shaking	4
quake	2
Thunder	1

[1] MyMemory: http://mymemory.translated.net/doc/spec.php
[2] Google Translate: http://translate.google.com

The system starts at the topmost level of the taxonomy and calculates for each node the classification score csc, i.e. the ratio of the number of identified terms that belong to the BoW of the node, and the total number of tweets in the respective cell and minute. A threshold of at least 0.3 is set to assure the relevance of the identified keywords in the current Twitter content. Assuming 20 tweets occurred in the cell in the last minute and the system yielded the results depicted in Table 2, the ratio for *Earthquake* would be 0.8 ($(10 + 4 + 2)/20$) and 0.05 ($1/20$) for *Meteorological* (the term "thunder" is in the BoW of *Hail/Thunder*).

Only the child node with the highest value is further analysed in an analogous manner. In case of two or more equal values as well as if all child nodes fail to reach the threshold, the parent node is set as type of the event. For example, if the system decided for *Hydrological* in the preceding level, but cannot distinguish between *Flood* and *Tsunami* based on the identified keywords, it will classify the event as *Hydrological*.

Figure 3: Disaster taxonomy with bag-of-words for the natural disaster type *Earthquake*



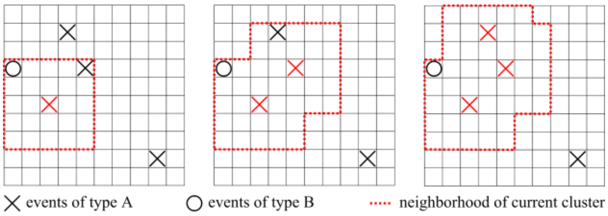
3.6 Spatial-thematic Clustering

For large scale natural disasters that exceed the area of a single cell, the system performs a spatial clustering to aggregate cells that represent the same natural disaster.

The system conducts an algorithm loosely based on the idea of the region growing method in image processing. Here, the initial seed points are the detected and classified single event cells in a time step. In contrast, to image processing, not all cells will be evaluated but only the set of seed points. Hence, in a 24-neighborhood around the seed points the systems searches for others of the same natural disaster type. The rather large neighbourhood tries to account for the inhomogeneous population distribution, which plays a major role in the detectability of events in a specific cell. Thus, even a high impact natural disaster can lead to a diffused detection of affected cells. Figure 4 shows an abstract, exemplary clustering process and the 24-neighborhood of a cell.

The information of the cells in a cluster is fused and from now on interpreted as one single event. Clusters with only one event cell are also referred to as clusters in the following.

Figure 4: Spatial clustering (abstract example); Red colour denotes events of the same spatial-thematic cluster



3.7 Temporal Monitoring

The temporal monitoring operates on the results yielded by the spatial-thematic clustering, i.e. it attempts to link the clusters from preceding time steps to the currently detected ones that refer to the same event.

Therefore, similar to the clustering process, the systems scans the database for detected events that were assigned to the same type of natural disasters and are falling in the merged neighbourhood of the current cluster. In case of a successful search, the associated tweets of the current cluster are persistently stored in relation to the ID of the existing event in the database.

In contrast, if the systems cannot link any existing event in the database to the current cluster, the cluster will be stored with its aggregated information and tweets as a new natural disaster with a unique event ID.

3.8 Notification and Visualisation

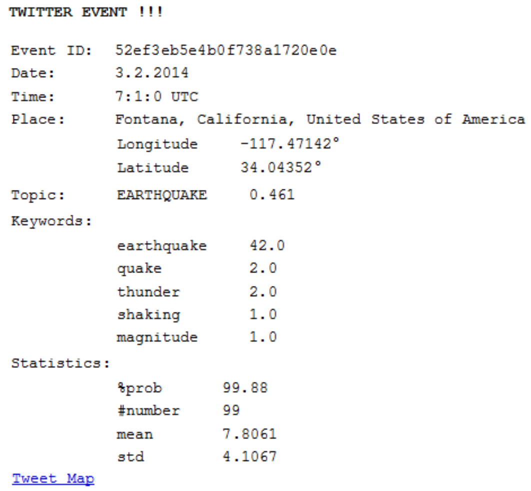
After the detection of such a new natural disaster, the system sends an automatic e-mail alert to a given address. The message contains the most important information of the event, such as date, time, geographic place and coordinates, the type of the natural disaster with its *csc*, the identified disaster keywords and their occurrences as well as the statistical values of the frequency analysis. These include the statistical probability p that the test decided correctly in favour of the alternative hypothesis, the number of tweets in the cluster and its corresponding mean and standard deviation for the minute of the day. Figure 5 shows the e-mail alert for an earthquake of magnitude 3.1 near Los Angeles.

The place is determined through an implemented call to the reverse geocoding service of OpenStreetMap (OSM). The geographical coordinates used as input parameters for the service are the longitude and latitude of the centroid of all tweets in the cluster.

At the end of the notification e-mail, a link to a *node.js* based web service is provided. By clicking on the link, a browser tab opens and visualises all tweets that were assigned to the specific disaster on a map. The basic map data also comes from OSM and is implemented with *leaflet.js*, an open-source JavaScript library for interactive maps.

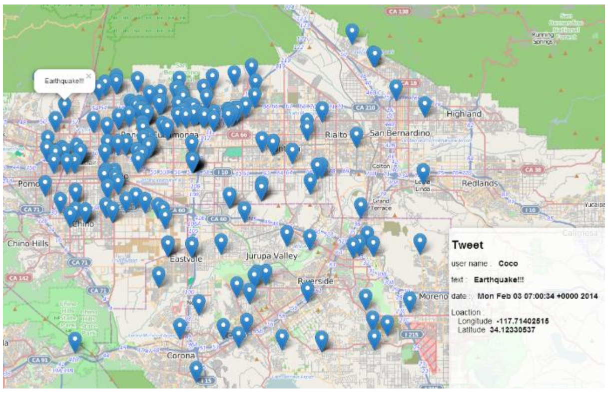
The web service communicates with the database to retrieve the corresponding tweets based on the unique event ID of the natural disaster depicted in Figure 5. The user can view the tweet information by clicking on the tweet markers. The screenshot in Figure 6 depicts the web service response for the earthquake in Figure 5.

Figure 5: E-mail alert with main information of the detected natural disaster


```
TWITTER EVENT !!!

Event ID:   52ef3eb5e4b0f738a1720e0e
Date:       3.2.2014
Time:       7:1:0 UTC
Place:      Fontana, California, United States of America
            Longitude     -117.47142°
            Latitude      34.04352°
Topic:      EARTHQUAKE     0.461
Keywords:
            earthquake     42.0
            quake          2.0
            thunder        2.0
            shaking        1.0
            magnitude      1.0
Statistics:
            %prob        99.88
            #number      99
            mean         7.8061
            std          4.1067
Tweet Map
```

Figure 6: Screenshot of the JavaScript based web service to visualize the event's tweets



Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © CloudMade

4 Experimental Results

Since 01/01/2014 the system automatically detected and analysed a total of 186 natural disasters of varying impact and different types. Table 3 shows the distribution on the different disaster types in absolute numbers and percentage. Due to the lack of a universal definition what constitutes a disaster, an evaluation based on a confusion matrix would not yield meaningful results. Therefore, absolute detection rates are not provided. However, since the actual aim of the system is to analyse natural disasters with impact on people, there is no information loss.

Depending on the disaster type's temporal characteristic the e-mail alert was sent within 40 seconds to 44 minutes from the beginning of the event. As expected, earthquakes are best suited for a detection in real-time (mostly below 2 minutes), because they have an exact starting time and occur unexpectedly. The 4.5 earthquake in Fontana, California on 15/01/2014 at 9:35:19 UTC caused an e-mail alert only 49 seconds later. The event had a statistical probability p of

96.8% and a classification score csc of 0.64. The system stored a total of 1791 tweets that are directly linked to the event containing mentions of "earthquake" (708), "quake" (66), "shaking" (42), etc. The last cluster detection that could be assigned to the event occurred at 10:35 UTC.

Table 3: Absolute number and percentage of automatically detected and analysed disaster types since 01/01/2014

type	number	percentage
Earthquake	78	41.9%
Hail/Thunder	43	23.1%
Natural disaster	18	9.7%
Meteorological	16	8.6%
Tornado	13	7.0%
Volcanic eruption	5	2.7%
Flood	5	2.7%
Tsunami	4	2.2%
Blizzard	2	1.1%
Drought/Heat wave	1	0.5%
Cyclonic storm	1	0.5%
Hydrological	0	0%

Volcanic eruptions have similar characteristics. However, they are rarely located very close to populated places and therefore their effects usually take longer to be noticed by the public. For example, the eruption of the Mt Kelud with its massive emission of ash in Java, Indonesia on 13/02/2014 at 15:50 UTC, was detected at 16:34 UTC.

Other disaster types such as *Flood* or *Hail/Thunder* have no discrete starting but evolve with time. Nonetheless, the system automatically analysed several such events. For example, the flooding caused by heavy rains, in parts of Jakarta, Indonesia on 29/01/2014 in the morning, was detected at 4:45 local time ($p = 99.9\%$ and $csc = 0.63$) and provided 288 tweets with on-site information (mentions: "flood(s)/ing" 84, "inundation" 28, etc.). The severe thunderstorm that hit New York City, USA in the evening of 13. February 2014 was detected at 20:45 local time and 786 tweets could be assigned to the event ($p = 99.4\%$ and $csc = 0.37$) with mentions of "thunder" (221), "lightning" (86), etc.

5 Outlook

The application, introduced in this paper, detects various natural disaster events such as earthquakes, floods or volcanic eruptions (cf. Section 4). The characteristics of the events, e.g. earthquakes and floods, are fundamentally different. On the one hand there is an abrupt punctual event and on the other hand there is a continuous and areal event, but both types are detected and analysed by the prototype. For a comprehensive and reliable evaluation, the approach has to be tested with every possible event type from the taxonomy (cf. Fig. 3). Additionally, the authors will expand the detectable event types on man-made disasters such as industrial accidents, building collapse or smog. Therefore, the hierarchical structure of the taxonomy has to be extended and the appropriate BoWs need to be compiled.

In a next step, the application will be incorporated into a framework for exploring all disaster related information from tweets. The long-term aim is to interpret the tweet's textual content in real time based on machine learning techniques to extract and classify relevant information. This information will help to improve the situational awareness for crisis management.

References

[1] Aggarwal, C., Zhai, C.: A Survey of Text Clustering Algorithms. In: Aggarwal, C.C., Zhai, C. (eds.) *Mining Text Data*, pages 77-128. Springer US 2012.

[2] Benson, E., Haghighi, A., Barzilay, R.: Event discovery in social media feeds. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, Association for Computational Linguistics, 2011.

[3] Dittrich, A., Lucas, C.: A step towards real-time detection and localization of disaster events based on tweets. In: *Proceedings of the 10th International ISCRAM Conference*, 2013.

[4] Gu, H., Xie, X., Lv, Q., Ruan, Y., Shang, L.: ETree: Effective and Efficient Event Modeling for Real-Time Online Social Media Networks. In: *Web Intelligence and Intelligent Agent Technology*, IEEE Computer Society, 2011.

[5] Lee, R., Sumiya, K.: Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection. In: *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, ACM, 2010.

[6] Metzler, D., Cai, C., Hovy, E.: Structured event retrieval over microblog archives. In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, 2382138 2012.

[7] Popescu, A.-M., Pennacchiotti, M.: Detecting controversial events from twitter. In: *Proceedings of the 19th ACM international conference on Information and knowledge management*, ACM, 2010.

[8] Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes Twitter users: real-time event detection by social sensors. In: *Proceedings of the 19th international conference on World wide web*, ACM, 2010.

[9] Sankaranarayanan, J., Samet, H., Teitler, B.E., Lieberman, M.D., Sperling, J.: TwitterStand: news in tweets. In: *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ACM, 1653781 2009.

[10] Walther, M., Kaisser, M.: Geo-spatial Event Detection in the Twitter Stream. In: Serdyukov, P., Braslavski, P., Kuznetsov, S., Kamps, J., Rüger, S., Agichtein, E., Segalovich, I., Yilmaz, E. (eds.) *Advances in Information Retrieval*, vol. 7814, pages 356-367. Springer Berlin Heidelberg 2013.