Earthquake emergency management by Social Sensing

Marco Avvenuti*, Stefano Cresci†, Mariantonietta N. La Polla†, Andrea Marchetti† and Maurizio Tesconi†
*Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Pisa, Italy,
Email: firstname.lastname@iet.unipi.it
†Istituto di Informatica e Telematica, Consiglio Nazionale delle Ricerche, Pisa, Italy
Email: firstname.lastname@iit.cnr.it

***Abstract*—Social Sensing is based on the idea that communities or groups of people provide a set of information similar to those obtainable from a single sensor. This amount of information generate a complex and adequate knowledge of one or more specific issues. A possible field of application for Social Sensing is Emergency Management. By using the Social Media it is possible to gather updated information about emerging situations of danger, in order to gain greater situational awareness and to alert interested parties promptly or verify information obtained through other channels. A system able to timely detect events that are of social concern can be referred to as an *Early Warning* system. In this work we propose a novel and general architecture for an early warning system and, as a proof-of-concept, we describe an implementation of this architecture for a real scenario. We use Twitter as source of information for the detection of earthquakes on the Italian territory. We compare our results with official data provided by the National Institute of Geophysics and Volcanology, the authority responsible for the monitoring of seismic events in Italy. Results show an high ability of the system in the timely detection of events with magnitude equal or greater than 3.5 Richter with only 10% of False Positives.**

I. Introduction

Social Media (SM) represent, in the context of Web 2.0, the most effective, sophisticated and powerful way to collect preferences, tastes and activities of groups of users [1]. Social Sensing is based on the idea that communities or groups of people can provide information data similar to those obtainable from a single sensor. This amount of information generates a complex and adequate knowledge of one or more specific issues[1]. Therefore, SM users can be treated as social sensors, namely as a source of information about situations and facts related to the users (e.g., their preferences or experiences) and their social environment. Among different fields of study regarding Social Sensing, the one about Emergency Management is very interesting. By using SM it is possible to gather updated information on emerging situations of danger. In this way, greater situational awareness can be gained, in order to promptly alert interested parties. In the case of an event or a disaster, the information needs to be collected as soon as possible. So doing more time is available to take correct decisions about the situation to deal with, and to timely notify the involved population and authorities. We refer to Early Warning Systems (EWS) as those information systems able to timely detect occurring events of social concern and thus able to deliver appropriated warnings [2].

As a source of information we aim to use spontaneous reports published by users on SM about dangerous events, such as natural disasters, both in small and large scale. The idea behind this study comes from the intuition that the first people who announce such phenomena are those directly involved in the event. The study is based on Social Networks (SN), a specific type of SM particularly suited to host this type of analysis due to their high level of interaction and the large number of users involved. The most popular SM with the highest number of subscribers in the world are two SN, Facebook and Twitter. The advantage of exploiting SN compared to traditional methods of investigation relies on the spontaneous participation of the users that in fact are not induced or guided in any way. In terms of SN characteristics, Twitter presents some peculiarities making it particularly suitable as a source of data for social sensing platforms. Twitter users generally talk about their activities and therefore of what is happening around them. Moreover, Twitter is more interactive and responsive than Facebook: due to the limitations imposed on the length of public messages (maximum 140 characters), the lifetime of a tweet is rather short, and therefore users are encouraged to tweet more frequently.

In particular, we will focus on reports related to earthquakes occurred in Italy. According to the Italian Civil Protection Dept.[2], during the last 2,500 years Italy was affected by more than 30,000 earthquakes of intensity greater than 4 Mercalli. As suggested in [3], the highly seismic nature makes Italy one of the best countries where to carry out a study on the detection of earthquakes based on Twitter data. On the other hand, the growing use of Twitter during the last years (in 2013 active users increased 40% in the world and 50% in Italy [25]) encourages and predicts a successful study.

The main goal of this work is to define a framework for the timely real-time detection of earthquakes. However, the proposed system is generally enough and easily adaptable to deal with other types of events. An important feature of our solution relies on its responsiveness. The detection of an event is not sufficient on its own but, instead, we aim to perform a timely detection in a real-time fashion. In particular, our system is able to detect an earthquake before the publication of the related news by official channels, such as official offices involved in the management of the events, newspapers and blogs [2]. Moreover, we implemented a simulator to test the system behavior on the whole dataset collected.

[1]http://beautifuldata.net/2013/01/social-sensors/

[2]http://www.protezionecivile.gov.it/

978-1-4799-2736-4/14/$31.00 ©2014 IEEE

The rest of the paper is organized as follows: Section II describes the related work. Section III details our proposal and Section IV describes the results obtained by applying our system on the Italian earthquakes scenario. Section V draws the conclusions and future work.

II. Related work

The use of SM for social sensing is currently the object of many studies. Several initiatives, both in the scientific and in application environments, has been developed with the aim of exploiting the information available on these platforms. In literature descriptive and general approaches are opposed to practical, sector-based experiences (e.g. epidemiology, earthquakes, etc.). One of the most interesting works concerns the prediction of the spreading level of H1N1 flu in the United Kingdom, starting from tweets [4] with an accuracy of 95%. The analysis was performed on a daily basis. The work proposed in [3] shares a similar goal with our system since it aims at the creation of an EWS for the real-time detection of earthquakes and tornadoes in Japan based on Bayesian statistics. The system was able to timely detect 67.9% (53 of 78) of the earthquakes with JMA scale 2 or more occurred in two months and to send event notifications with delays ranging from 20 seconds to a minute. Data acquisition is performed via the Twitter Search API which accesses only a portion of all the tweets produced which can result in possible limitations for the event detection. A similar work is EventRadar [5] that proposes a novel method for the detection of local events (involving a limited number of people) using a temporal and spatial analysis in real-time and reached an accuracy level of 68%. The work presented in [6] describes approaches based on Burst Detection generally applied to the detection of topics in data streams.

From an application point of view, one of the most interesting initiatives is represented by Emergenza24[3], the experimental version of the "Social Network for Emergency Management", for the Italian territory. Similarly, the SMEM platform (Social Media Emergency Manager)[4] tries to promote the use of an hashtag, *#smem*, to report emergencies or other events of social interest. As a generic approach, we can mention the European initiative Alert4All[5] that aims to create a framework to improve the effectiveness of warning messages and communications with the population in case of disasters and focuses on the role of SM in the emergency communications [7], [8]. The SMART-C project [9], [10] describes an high-level, multi-modal framework able to collect and integrate data from different sources such as Social Networks, blogs, telephone land line communications, SMS, MMS. The SHIELD system [11] was designed and developed to exploit mobile devices in order to reduce the response and rescue times for victims of micro-criminality in the U.S. university campuses. City Sourced[6] is a web and mobile application, widespread in some cities in the U.S., that allows citizens to send reports of discomfort or emergencies at urban level.

The analysis of the literature has highlighted the lack of a comprehensive approach to the problem of Emergency Management on SM. What is missing is an architectural approach that both details the necessary components and addresses the general and practical problems of typical applications.

[3] http://www.emergenza24.org/

[4] http://www.socialmediaemergencymanager.com

[5] http://www.alert4all.eu/

[6] http://www.citysourced.com/

III. Proposed solution

In order to design a system for the detection of events, firstly we must identify the characteristics of the events to be analyzed. As stated in [5] events can be initially divided into real and virtual events. Virtual events are phenomena that occur just in the online world. Real events can be further divided into local and global events. A local event has a specific location both temporally and geographically, while global events represent events such as global celebrations and holidays (Christmas, Valentine's Day, etc.). Local events differ in the number of users and/or the geographical area involved and this can affect the ability to detect them. Since we exploit people as sensors, events that involve fewer people are more difficult to be detected. Moreover, as stated in [5], the development of a system that can manage both types of events, those of small and medium-large scale, is a challenging task. As previously stated we will focus our attention on the detection of seismic events, starting from tweets. The detection process is useful for triggering a more accurate study of the event in order to assess damages and impact on population and because of the lack of institutional monitoring systems in some areas of the world.

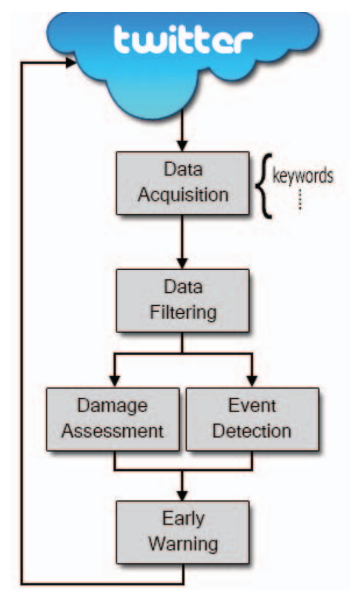
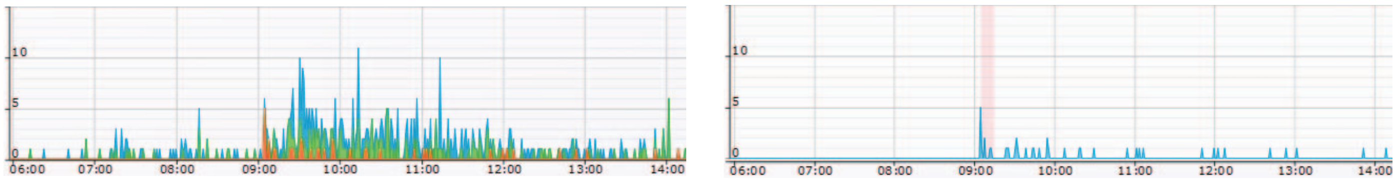


Fig. 1: Proposed architecture

Taking into account previous experiences, we designed an architecture composed by the following modules, as graphically highlighted in Figure 1:

- Data Acquisition: collection of Twitter data;
- Data Filtering: reduction of the noise caused by messages collected but not related to an ongoing event;
- Event Detection: detection of an event starting from the analysis of filtered data;
- Damage Assessment: estimation of the consequences of the event;
- Early Warning: publishing of the news related to the events;



(a) *Before* filtering (b) *After* filtering

Fig. 2: Impact of the filtering phases on the number of analyzed tweets.

A. Data Acquisition

The role of this module is very important because the system only operates on data collected in this phase. Errors at this stage, especially regarding the loss of data, have to be minimized since they will propagate throughout the system impairing the ability to detect events. In order to make this module as independent as possible from the source platform (eg, Twitter, Facebook, etc.), collected data are stored in a common format. We can split the process of data acquisition in (i) the selection of keywords and (ii) the use of the interface provided by the SN for the collection of the data. Obviously, only a small amount of the global production of messages on the SN is related to a specific event type. In order to filter messages produced on Twitter (9100 new tweets per second in the average) and to collect only those interesting to our goal, it is necessary to identify some keywords related to the event type (the choice of the keywords is detailed in section IV-A). In general, the name of the event and its synonyms (eg, earthquake) could represent a good starting point.

Among the methods provided by Twitter for information extraction, we exploit the Streaming API, which opens a persistent connection with a stream of tweets. By using this connection, new tweets that contain the selected keyword(s) can be collected. In contrast with the Search API used in the study described in [3], which gives access only to a subset of all the tweets produced, the Streaming API potentially allows us to capture all the tweets matching the search criteria. Tweets published before the opening of the connection can not be retrieved, but this limitation is not a problem for our system as we are interested in the real-time detection of events.

Another limitation can arise considering that, as the connection can potentially access the entire flow of tweets produced in the world, Twitter delivers at most 1% of the total traffic and automatically cuts off the excess, indicating the number of dropped tweets. However, our system never suffered from such a limitation over a two months long experiment, during which the collected tweets generated a traffic never exceeding the 1% threshold. Falling-behind is another concern: after receiving the tweets from the API, clients have to process them rapidly, because during this processing the client cannot accept other tweets. To address this problem, tweets are stored in ad-hoc data structures to be processed. To guarantee the robustness and the reliability of the system we also implemented additional mechanisms that manage rate-limit and generic connection problems in the use of the APIs.

B. Data filtering

Using keywords to query the platform allows us to gather messages potentially related to an event. However, not all the messages gathered in this process talk about an ongoing event. Some messages can be misleading for the Event Detection module and must be filtered out as noise. We identified two different sources of noise: (i) messages in which the keyword is used with a different meaning from the one related to the searched event and (ii) messages in which the keyword refers to a past event.

The Data Filtering module reduces noise by cleaning data in 2 steps. A pre-filtering phase discards tweets that are retweet messages, reply messages, messages sourced by a list of 345 accounts of official channels that periodically publish information about seismic events, and messages containing words listed in a blacklist.

After pre-filtering, a more sophisticated filtering phase is performed by a classifier that infers the class of a tweet starting from a model. We used the Weka [12] tool to train and produce our classifier. During the offline training phase, the classifier was trained using two distinct sets of messages, tweets related and tweets not related to a seismic event in progress, respectively. Tweets of the training-set were manually classified using an ad-hoc interface. The training-set contained 1412 tweets, equally distributed in two classes: *useful* and *not useful*. Starting from the content of tweets, we define a set of features related to the number of entities in the text: URL, mentions, words, character, punctuation and slang/offensive words. We then associate to each tweet a set of related features. The list containing the features and the specification of the class of the tweet was then processed, using Weka, in order to obtain a model that defines the class of the tweet based on its attributes. The prediction is performed at run-time by invoking the classifier every time a pre-filtered tweet is received. As Weka generally needs less than a second to predict the class of a new tweet, we can use the fine-grained classifier filter in real-time applications.

Figure 2 shows the comparison between the numbers (y axis) of all collected tweets (Figure 2a) and the tweets that passed the filtering phase (Figure 2b) during a seismic event of low magnitude. The earthquake occurred in Modena, Italy, September the 4th, 2013 (magnitude 3.3 Richter) at 09.03 a.m. The peak in the collected tweets starts around 09.04 a.m.. The blue plot is related to all the collected tweets. The green plot shows the tweets that passed the pre-filtering phase and, finally, the red plot represents the tweets after the filtering phase with the classifier.

C. Event Detection

In this phase events are detected according to different measures. The novelty of our solution lies on the multi-level nature of the analysis provided on the collected data. The detection of an event is the result of following analysis: (i) a temporal analysis, for the real-time detection of the event, and (ii) a spatial analysis, to infer the event location. A deeper discussion about the algorithm used to perform the event detection is out of the scope of this paper.

1) Temporal Analysis: The occurrence of an event is detected by observing an unexpected growth in the number of messages related to a category of events. A key point in this phase is the identification of the minimum number of messages needed to state that an event has occurred. This number represents a trade off between the sensitivity of the system (a small number of messages is required) and the remaining noise after the filtering process. To address the problem of detecting events, we investigated a number of techniques, such as the realization of a temporal model based on Bayesian statistics [3], the use of Peak-Detection algorithms, the Corrected Conditional Entropy (CCE) [13] and different change-detection and burst-detection algorithms. In our solution we used a novel burst-detection approach. A burst is defined as the occurrence of a large number of events within a time window [14]. The detection is based on the calculation of the frequencies of data in a time window. To locate a burst we compare the current frequency with a reference frequency calculated as the average frequency per minute of the total number of tweets and the number of tweets containing the keyword(s) published in the last week. Figure 3 shows the arrival times of filtered tweets during an earthquake occurred at 00:09 a.m, August 27, 2013 in Umbria and Marche regional districts. After T1, the occurrence time of the earthquake, a big burst of tweets was recorded by our system.

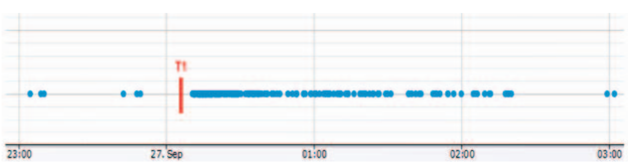


Fig. 3: A burst of tweets registered after an earthquake (occurred at T1)

2) Spatial Analysis: A key information to completely describe an event is its spatial location. Twitter offers the possibility to automatically geocode messages, but unfortunately georeferenced tweets collected in our study were only 1.5% of the total. Considering only the georeferenced messages would drastically reduce the probability to detect small-scale events. To solve this problem, in [3], a solution is proposed based on the location of the accounts of Twitter users. If a tweet is not georeferenced, the position of the event is inferred from the position of the twitter account. This solution is not reliable. Twitter asks users to fill the location field in the registration process but this is not mandatory. Even if the field is filled, there is no control over it and users can type whatever they want. Instead we propose a solution based on the analysis of the content of messages. We extract names of places from the text of the messages and we geocode them. To accomplish this task we make use of TagMe, a service of text disambiguation developed at the University of Pisa [15]. This tool can be specifically set to work with tweets and outputs a list of tags with the description of the terms and other related information. Furthermore, we then extract the coordinates associated to the places disambiguated by TagMe via SPARQL [16] queries to DBpedia [17].

D. Damage Assessment

The Damage Assessment (DA) is the process that allows emergency management personnel to determine the impact and the consequences of an emergency on communities and infrastructures. Typically, the DA process takes place in the aftermath of an emergency and involves specialized personnel to assess the consequences visiting the location of the event. We plan to use information shared by people directly involved in the event to perform DA automatically. To accomplish this task a key point is the timely detection of the event. In this way we can expand the set of keywords adding those terms related to the location of the event. We also add to this set the most frequent words in the tweets already collected in order to gather more information about the detected event. This augmented set is used to perform the damage assessment task exploiting both the textual descriptions and the multimedial content (eg: videos, photos). The keywords used in the damage assessment process should be as generic as possible, not related to a specific type of event, but that can be related to the possible consequences of an emergency in terms of damages to infrastructure and/or people involved.

Fig. 4: Example of a tweet carrying sensitive information for the DA task

Figure 4 shows a very interesting tweet reporting a short description in Italian language and a photograph of a landslide occurred as a consequence of the earthquake that struck Ancona at 08:44 a.m, August 22, 2013.

IV. Testing and Results

For the validation of results, we used official data published by the National Institute of Geophysics and Volcanology (INGV), the authority responsible for the monitoring of seismic events in Italy. INGV uses different channels, including Twitter, to spread detailed information about all seismic events, detected by their equipment, having magnitude equal or greater to 2 on the Richter scale. Using the account @INGVterremoti, this authority provides for each detected event: magnitude (type and value), date and time of the earthquake, geographical coordinates of the epicenter, depth, administrative province of Italy, link to a page with a detailed description of the earthquake.

A. Collected Data

The Data Acquisition module was active for more than two months, from July 19, 2013 to September 27, 2013. In order to select a comprehensive set of keywords we started both from terms reported in the literature [3] and other words related to earthquakes in the Italian language. We progressively restricted the initial set of 9 keywords eliminating the ones that didn't show a correlation between their frequency of usage and the seismic events reported by official agencies. Furthermore, we discarded those keywords, such as "crollo" (wreckage) and "crepa"(crack), specifically related to the task of Damage Assessment; those keywords, such as "sisma" (seism) and "magnitudo" (magnitude), that are often used in official communications rather than in spontaneous user messages; and those keywords, such as "trema" (shakes) and "tremando" (shaking), that are too generic and used in contexts other than the events we want to detect. At the end of this selection process we figured out that the most selective Italian words to collect tweets about ongoing earthquakes are "terremoto" (earthquake) and "scossa" (tremor).

For the category of seismic events we collected 64,878 tweets (926 tweets per day in the average). The filtering process eliminated up to the 88% of the collected tweets. These figures helped to better understand the phenomenon of noise that affects the channel and the importance of the data filtering phase. The average load on the system was less than 1 tweet per minute, while the highest peaks reached 5 tweets per second and never lasted for more than 3 seconds. During the experiment, our system has always been able to meet its real-time requirements. The entities extracted from the text of the tweets include photos and videos (media type), URLs, hashtags and mentions. We collected up to 1.9 million entities and data about more than 330,000 different users.

A total of 1412 tweets have been manually classified to build the training set for the classifier used in the Data Filtering module. We tested different training algorithms available in Weka: the best result led to an accuracy of 90.085% (1272/1412 tweets) and was obtained using the decision tree J48, corresponding to the Java implementation of the C4.5 algorithm [18] with a 10-fold cross validation [19]. The training phase results are reported in the confusion matrix of Table I, where columns represent the instances in the predicted class and rows represent the instances in the actual class.

		Predicted Class	
		Useful	Not Useful
Actual Class	Useful	654	52
	Not Useful	88	618

TABLE I: Results of the training phase

Collected data were stored in a database and used as a validation set in an off-line simulative study. The use of the simulator allowed us a comprehensive analysis of the entire dataset in less than 3 hours.

B. Results evaluation and analysis

We cross-checked the events detected by the simulator against the official reports by INGV. We classified results as in the following:

- True Positives (TP), events detected by the system and confirmed by INGV;
- False Positives (FP), events detected by the system, but not confirmed by INGV;
- False Negatives (FN), events reported by INGV but not detected by the system.

and we used the following evaluation metrics:

- Precision, ratio of correctly detected events among the total number of detected events:

$$\mathit{Precision} = \frac{TP}{TP + FP} \tag{1}$$

- Recall, ratio of correctly detected events among the total number of occurred events:

$$\mathit{Recall} = \frac{TP}{TP + FN} \tag{2}$$

- F-Measure, harmonic mean of Precision and Recall:

$$\mathit{F\text{-}Measure} = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

True Negatives are meaningless, as it would mean counting the number of earthquakes that did not happen and that our system did not detect.

Table II reports results obtained analyzing data collected from 2013-07-19 and 2013-09-23. The magnitude reported in the table is the official value stated by INGV and is not calculated by our system.

Magnitude	Occurred Events	TP	FP	FN	Precision	Recall	F-Measure
> 2.0	403	16	41	387	28.07%	3.97%	6.96%
> 2.5	101	15	41	86	26.79%	14.85%	19.11%
> 3.0	25	12	18	13	40%	48%	43.64%
> 3.5	10	8	3	2	72.73%	80%	76.19%
> 4.0	6	4	0	2	100%	66.67%	80%
> 4.5	1	1	0	0	100%	100%	100%

TABLE II: Results of detected events

The analysis evidenced that a timely detection of events with magnitude lower than 3 Richter is very difficult. This is because the majority of these earthquakes are only felt by equipment and not by people. For events with magnitude equal or greater than 3.5 Richter, results show a good performance of the system in terms of F-Measure. This is especially remarkable considering that events of magnitude around 3 Richter are generally felt only by a very small number of people. In our study, for such events we always registered less than 10 tweets per event.

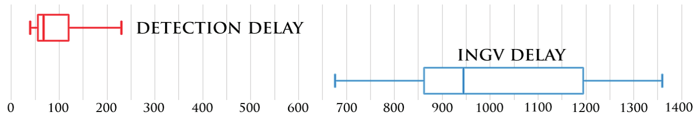


Fig. 5: Detection delays compared to INGV reports

As previously stated a key factor in the evaluation of an EWS is represented by the detection delays. We compared detection delays by our system with the delays of the official reports by INGV. Figure 5 shows the results of this comparison by means of box-plot distributions expressing delay times in seconds. The top plot (red colored) is related to our system while the bottom one (blue colored) to INGV reports. Noticeably the median of the detection delays of our system is in the region of 1 minute while the median of INGV delays is over 15 minutes.

An issue which arose in this work and which is still not addressed is related to the number of FPs detected by the system. Especially for low magnitude earthquakes, FPs are a significant percentage of the total of the events. The scaling value of the FPs is due to the fact that in case of an event of high magnitude (e.g. > 4.0), the number of sensors increases. This increment corresponds to an increased dimension of the tweet set related to the event. It is reasonable to conclude that these FPs are caused by the remaining noise on the channel.

Despite the prototypical nature of the system and the small size of the statistical sample, the results obtained are better than those achieved by other recently published works.

V. Conclusions and Future Work

This work presents a general application framework for Early Warning Emergency Management. The implemented prototype demonstrates the effectiveness of our approach in the field of seismic events, but the proposed architecture is general enough to be applied to other application contexts, such as wildfires, traffic jams, landslips, floods. The results achieved by the prototype are overall better than those reported in similar works available in the literature. In particular, the Data Acquisition module proved to be robust and reliable. During the implementation, some critical issues requiring additional investigations have emerged. Among these, the task of data filtering and the task of event detection are particularly relevant. In this work we didn't take into account security concerns which can arise if groups of people collude to generate fictitious tweets referring to an earthquake. The adoption of security mechanisms, such as fake accounts detection, is reserved for future work.

An in-depth analysis of the results leads to a general conclusion related to the characteristics of social sensors. Social sensors are not uniformly distributed throughout the territory and naturally cluster in cities. This results in an increased difficulty in the detection of events that take place in sparsely populated areas. Furthermore, our sensors show a much lower sensitivity at night; a more intense event is necessary in order to trigger reactions, i.e. tweets, from social sensors during night hours.

References

[1] A. Rosi, M. Mamei, F. Zambonelli, S. Dobson, G. Stevenson, and J. Ye, "Social sensors and pervasive services: Approaches and perspectives," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 525–530.

[2] N. Waidyanatha, "Towards a typology of integrated functional early warning systems," *International Journal of Critical Infrastructures*, vol. 6, no. 1, pp. 31–51, 2010.

[3] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 851–860.

[4] V. Lampos and N. Cristianini, "Tracking the flu pandemic by monitoring the social web," in *Cognitive Information Processing (CIP), 2010 2nd International Workshop on*. IEEE, 2010, pp. 411–416.

[5] A. Boettcher and D. Lee, "Eventradar: A real-time local event detection scheme using twitter stream," in *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*. IEEE, 2012, pp. 358–367.

[6] R. Ebina, K. Nakamura, and S. Oyanagi, "A real-time burst detection method," in *Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on*. IEEE, 2011, pp. 1040–1046.

[7] S. Nilsson, J. Brynielsson, M. Granåsen, C. Hellgren, S. Lindquist, M. Lundin, M. Narganes Quijano, and J. Trnka, "Making use of new media for pan-european crisis communication," in *Proceedings of the Ninth International Conference on Information Systems for Crisis Response and Management (ISCRAM 2012)*. ISCRAM, 2012.

[8] F. Johansson, J. Brynielsson, and M. N. Quijano, "Estimating citizen alertness in crises using social media monitoring and analysis," in *Intelligence and Security Informatics Conference (EISIC), 2012 European*. IEEE, 2012, pp. 189–196.

[9] N. R. Adam, B. Shafiq, and R. Staffin, "Spatial computing and social media in the context of disaster management," *Intelligent Systems, IEEE*, vol. 27, no. 6, pp. 90–96, 2012.

[10] N. Adam, J. Eledath, S. Mehrotra, and N. Venkatasubramanian, "Social media alert and response to threats to citizens (smart-c)," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2012 8th International Conference on*. IEEE, 2012, pp. 181–189.

[11] G. S. Thakur, M. Sharma, and A. Helmy, "Shield: Social sensing and help in emergency using mobile devices," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–5.

[12] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[13] A. Porta, G. Baselli, D. Liberati, N. Montano, C. Cogliati, T. Gnecchi-Ruscone, A. Malliani, and S. Cerutti, "Measuring regularity by means of a corrected conditional entropy in sympathetic outflow," *Biological cybernetics*, vol. 78, no. 1, pp. 71–78, 1998.

[14] X. Zhang and D. Shasha, "Better burst detection," in *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*. IEEE, 2006, pp. 146–146.

[15] P. Ferragina and U. Scaiella, "Fast and accurate annotation of short texts with wikipedia pages," *Software, IEEE*, vol. 29, no. 1, pp. 70–75, 2012.

[16] J. Pérez, M. Arenas, and C. Gutierrez, "Semantics and complexity of sparql," *ACM Transactions on Database Systems (TODS)*, vol. 34, no. 3, p. 16, 2009.

[17] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "Dbpedia-a crystallization point for the web of data," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 3, pp. 154–165, 2009.

[18] J. R. Quinlan, *C4. 5: programs for machine learning*. Morgan kaufmann, 1993, vol. 1.

[19] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.