User-Assisted Information Extraction from Twitter During Emergencies

Zoha Sheikh
National University of Sciences and Technology, Islamabad, Pakistan
14msitzsheikh@seecs.edu.pk

Hira Masood
National University of Sciences and Technology, Islamabad, Pakistan
14msithmasood@seecs.edu.pk

Sharifullah Khan
National University of Sciences and Technology, Islamabad, Pakistan
sharifullah.khan@seecs.edu.pk

Muhammad Imran
Qatar Computing Research Institute, HBKU
Doha, Qatar
mimran@hbku.edu.qa

ABSTRACT

Disasters and emergencies bring uncertain situations. People involved in such situations look for quick answers to their rapid queries. Moreover, humanitarian organizations look for situational awareness information to launch relief operations. Existing studies show the usefulness of social media content during crisis situations. However, despite advances in information retrieval and text processing techniques, access to relevant information on Twitter is still a challenging task. In this paper, we propose a novel approach to provide timely access to the relevant information on Twitter. Specifically, we employee Word2vec embeddings to expand initial users queries and based on a relevance feedback mechanism we retrieve relevant messages on Twitter in real-time. Initial experiments and user studies performed using a real world disaster dataset show the significance of the proposed approach.

Keywords

social media, disaster response, query expansion, supervised learning

INTRODUCTION

The adaptation of microblogging platforms such as Twitter during crises, emergencies, and time-critical events has increased recently. Easy access to social networks provides ways to produce and retrieve information in different forms, such as textual messages, images, and videos. For instance, at the onset of a crisis event, people use social media platforms to fulfill their quick information needs. Access to critical information becomes more important, especially in the first few hours, when no other information sources such as, traditional media, news channels. are available. Moreover, rapid access to important information can help humanitarian organizations gain situational awareness, early decision-making, and to launch relief efforts accordingly.

Past studies have shown the presence of various types of important information on social networks (Hughes and Palen 2009; Starbird et al. 2010; Imran, Castillo, Diaz, et al. 2015). For instance, this includes information about critical infrastructure damage, reports of injured or dead people, urgent needs of those affected, reports of missing people, donation requests or offers and so on. Moreover, studies have shown the usefulness of such online information for disaster response and management. However, filtering a deluge of noisy information during an event to retrieve relevant information is a non-trivial task. Different approaches based on automatic, crowdsourcing, or combination of both have been proposed to process social media content during an event. For instance, Artificial Intelligence for Disaster Response (AIDR) (Imran, Castillo, Lucas, et al. 2014) is one such application that uses supervised classification techniques to process millions of messages posted on Twitter. However, training supervised machine learning classifiers at the onset of an event requires human-labeled data. Obtaining human-labeled data in the early hours can be done either via paid crowdsourcing (e.g. using CrowdFlower[1] or Amazon

[1] http://crowdflower.com/

Mechanical Turk[2]) or by employing volunteers. In either case, one has to pay a cost, which is *monetary* in the case of paid crowdsourcing and *time* in the case of volunteers. Ultimately, the scarcity of labeled data pose challenges for machine training and consequently it delays the processing of important information for end-users.

While obtaining labeled data is a time consuming task, but at times of a crisis there are usually abundant unlabeled data. For example, during the 2012 Sandy hurricane, the highest peak observed was around 16k tweets/min posted on Twitter. Given, we have no labeled data but access to thousands of unlabeled tweets in the first few hours when a disaster occurs, in this work, we aim to achieve the following two objectives:

(i) To fulfill end-users' critical information needs at the onset of a crisis situation when training a supervised classification system is impossible due to the scarcity of labeled data.

(ii) While fulfilling users' information needs, the second objective is to train a supervised classification system as quickly as possible to enable automatic prediction of new tweets.

To meet the above two goals, in this work, we propose a novel approach that utilizes state-of-the-art techniques from both unsupervised and supervised machine learning fields. Specifically, for the first task, we use a special form of word representation, called "word embeddings". We use the word2vec implementation from Mikolov et al. 2013 and train our own word2vec model using crisis-related tweets collected during a real disaster event. Given a user query, which usually consists of three to five words, we use the trained word2vec model to expand the original query terms using an automatic query expansion technique. The query expansion concept has a long history in information retrieval and has been applied in a number of contexts and application areas of information retrieval (Carpineto and Romano 2012). Basically, for each term in the query, we find top-n contextually similar words that we learn from a large corpus. The expanded query is then used to retrieve top-n tweets that have high similarity with the query vector. Moreover, we optimize the initial query. To get an optimal query vector, which can increase the retrieval of relevant tweets, we use the Rocchio algorithm (Joachims 1996).

To achieve the second objective, that is to train a machine learning classifier while retrieving relevant documents to fulfill the user's information needs, we adapt a relevancy feedback mechanism to get user's feedback on query results. In a relevance feedback technique, users of a system are involved in the retrieval process to improve the system's ability to fetch relevant results over time (Ghorab et al. 2013). In our case, the user provides feedback whether a returned result is relevant or not. We not only use this feedback to optimize the query vector, but also consider it as gold standard labels. That means, user's agreement on items relevancy is used to label those items accordingly. Upon collecting a handful of such labeled examples, we train a supervised machine learning classifier. Subsequent relevancy prediction is then performed by the trained model. The relevancy feedback mechanism continuously runs and keeps improving the classification performance. The results obtained from the experiments and user studies show the significance of the proposed approach. Once deployed, people in the crisis zone, affected communities, and crisis managers will be the users of this system.

The rest of the paper is organized as follows. In the next section, we provide detailed description of our proposed approach. Details regarding datasets and experiments are provided in the dataset and experiment section. Next, we summarize related work and conclude the paper in the last section.

PROPOSED APPROACH

To retrieve relevant tweets against a user query during an on-going crisis event, we propose an approach which mainly consists of two major computational components. The first component responsible for query expansion and fetching of relevant tweets, whereas the second component implements the relevance feedback mechanism and query optimization. Before we get into the details of the system, we formally define our task as follows:

Given a query q and a large set of unlabeled tweets T_{ul}, our task is two-fold: (i) to fetch the most relevant tweets T_r from the unlabeled tweets set T_{ul} by expending the query q using an already trained word embedding model (Let U be an $|V| * k$ term embeddings matrix); (ii) given a training data (x_i, y_i), where $x_i \in X$ and $y_i \in (1, ..., K)$ class labels (in our case two labels), train a machine learning predictor $f : X \rightarrow Y$, where X is the set of documents (tweets in our case) that we want to label to one of the Y labels.

Figure 1 shows the high-level pipeline of our system. The system receives a query from the user. In our case, a query usually consists of three to five words, e.g., "reports of inured or dead people". A straight-forward approach to get related tweets is to run a similarity (e.g., cosine similarity function) between a query and unlabeled tweets. Due to limited context given in such short queries, the coverage of the returned results will be very low thus the retrieved documents will either be too few or irrelevant. To resolve this issue, we expand the context of the query.

[2]http://mturk.com/

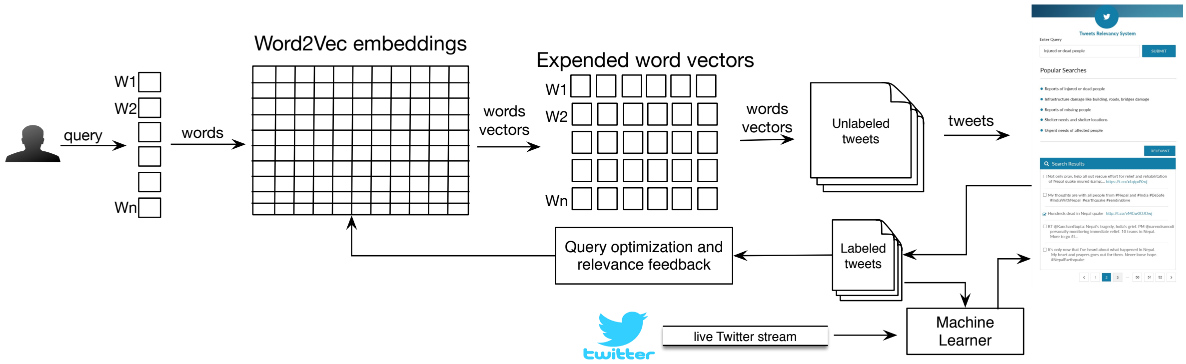

Figure 1. Query expansion and relevance feedback pipeline to retrieve relevant tweets and machine training

For this purpose, we use a state-of-the-art recent approach for word representation known as "word embeddings". Mikolov et al. 2013 proposes two different models (CBOW and skip-gram). In this work, we use the skip-gram model, which given a large corpus of text it learns the proportions of word occurrences and their context (e.g., five words before and after a target word). So, then given a window size n, words around a target word w, the skip-gram model predicts the neighboring words.

We trained a skip-gram based word2vec model using approximately 1200k crisis tweets with vocabulary $v = 108,025$ along 300 dimensions using a window size $n = 3$. We used the 2015 Nepal earthquake crisis-related tweets from CrisisNLP (Imran, Mitra, and Castillo 2016). The trained model is then used for query expansion. From the user query, we first remove stop words if any and then get top-20 highly contextually/semantically similar words from the trained word2vec model. Each query word is expended along 20 contextually similar dimensions. In figure 1, we show this representation under expended word vectors.

The original query words along with expended word vectors are then used to retrieve high similar tweets from the unlabeled tweets corpus. Specifically, we first take an average over the expended words vectors and then by using cosine-similarity measure we fetch top-20 most similar tweets. The retrieved tweets are then presented to the user. The user examines the tweets and mark the ones which he/she thinks are relevant to the original query. The user annotation/marking procedure divides the result set into two sets (i) relevant set (i.e., the tweets the users marked as relevant), (ii) irrelevant set (i.e., the remaining tweets the user did not mark).

Now, given the two sets of tweets from users (relevant and irrelevant), we aim to optimize the expended query vector so to get good results during the next iteration. For this purpose we use the Rocchio algorithm (Joachims 1996), according to which the optimal query vector is the one that maximizes similarity to relevant documents while minimizing similarity to irrelevant documents. Formally, it can be defined as:

$$Q_{opt} = argmax[sim(Q_{org}, D_r) - sim(Q_{org}, D_{nr})]$$

where:

Q_{opt} represents optimal query vector
Q_{org} represents original query vector
D_r represents relevant documents
$D_n r$ represents non-relevant documents

Using the relevance feedback mechanism, the optimal query optimization runs until it converges (i.e. an optimal query is obtained). The number of iterations required to get an optimal query depends on a number of factors. For instance, the complexity of the query (simple or complex queries) or on the length of the query (short or long). All such factors effect the overall retrieval procedure and the time to get an optimal query.

While running the relevance feedback and performing the query based tweets retrieval, we keep track of the labeled tweets. Once a handful of labeled tweets are collected, in our case 50 tweets in each category, the machine learner trains a relevancy classifier. We treat this as a two class classification problem i.e. relevant and irrelevant. In this case, the relevant class represents the tweets relevant to the user query, which are marked by the user itself. As our learning scheme, we use Random Forest with 100 trees. The model is trained using uni-gram and bi-gram based features extracted from both the relevant and irrelevant sets and we select top 1k most informative features.

The model evaluation is performed using 10-fold cross-validation technique. Evaluation results from various experiments are presented in the next section.

Once a model is trained, automatic categorization of subsequent tweets from Twitter live stream starts. The user seamlessly start receiving tweets categorized by the classifier. At this point, the query based tweets retrieval procedure stops. However, the feedback mechanism can still be running. But, now the user feedback is used to improve the existing trained model instead of query optimization. We train new models upon receiving a batch consisting of 50 labeled tweets. We recommend users to frequently mark results (relevant or irrelevant), when they observe high misclassification from the system.

DATASET AND EXPERIMENTS

We use the 2015 Nepal earthquake data from CrisisNLP (Imran, Mitra, and Castillo 2016). The dataset was collected from the Twitter streaming API using different keywords and hashtags during the disaster. In our case, we use around 1,200k unlabeled data. Before training the word2vec model and classifiers, we perform data pre-processing. Specifically, we change tweets to lower case, remove links, stop words, mentions, RT words, special characters, and extra spaces. We used the following five queries to retrieve the relevant tweets using our query expansion and optimization techniques.

(i) Reports of injured or dead people
(ii) Infrastructure damage like building, roads, bridges damage
(iii) Reports of missing people
(iv) Shelter needs and shelter locations
(v) Urgent needs of affected people

In below, we show a list of top five tweets retrieved by our system for each query after query optimization.

Reports of injured or dead people

T1: `News1130radio: Hundreds confirmed dead after massive #earthquake in #Nepal URL`
T2: `Massive 7.9 magnitude earthquake strikes Nepal, 108 feared dead`
T3: `#BreakingNews Death toll from Nepal quake nears 2,000 URL`
T4: `My heart is with those affected by this powerful quake. 7.8 Earthquake in Nepal Kills Hundreds, Levels Buildings`
T5: `abcnews: Nepal Quake: search for survivors, with 50 people missing in Dharahara Tower collapse URLs`

Infrastructure damage like building, roads, bridges damage

T1: `Several buildings including ancient temples collapse in Kathmandu after magnitude 7.9 quake URL`
T2: `Nepal's earthquake damaged a 19th century building. Sad. #NepalEarthquake`
T3: `More than 6.6 mil people in the area affected by #Nepalquake. Widespread damage and destruction of infrastructure feared`
T4: `Buildings Are Down And Roads Are Out After Major Nepal Earthquake, Cnn Sister Network Cnn-Ibn A powerful`
T5: `monuments decimated, roads and bridges destroyed, electricity out and general chaos #Nepal needs the world's support`

Reports of missing people

T1: `Rescue teams raise efforts on mount Everest to search for missing climbers, 30 climbers are injured, rescue continues..`
T2: `googleindia: We've just launched a Person Finder instance to help track missing persons for the #Nepal earthquake`
T3: `pray for nepal, everest.. avalanche, hikers are missing`
T4: `Trekkers reported a major avalanche on Mount Everest, with some teams reported missing.URL`
T5: `Nearly 700 people known dead, many more missing under rubble, as Mag 7.8 #Earthquake hits near #Kathmandu..`

Shelter needs and shelter locations

T1: `Stay strong Nepal. #Kathmanduquake. Send people to these shelters closest to you. URLs`
T2: `RT sarojdhakal: Make your shelter and help build others #NepalEarthquake #earthquake URL`
T3: `PLAN staff reporting collapsed and damaged buildings in Kathmandu. Shelter needed as buildings unsafe #NepalQuake`
T4: `Major aftershock hits Nepal after earthquake Habitat for Humanity mobilizing to assist with shelter needs`
T5: `brick school being used as shelter for villagers who've lost homes #NepalQuake #Wales #par`

Urgent needs of affected people

T1: `The Govt. and people of kathmandu urgent need the medicine supply and small scale mchines tools for removing collapsed building.`
T2: `RT CHOICEorg: We are encouraging #donations that will be directed to the most urgent needs of #Nepal`

Query	Precision	Recall	F1	# of relevant examples	# of irrelevant examples
Q1	0.67	0.82	0.73	81	315
Q2	0.82	0.91	0.86	47	253
Q3	0.94	0.91	0.92	27	213
Q4	0.91	0.80	0.85	13	227
Q5	0.85	0.92	0.88	36	144

Table 1. Classification results in terms of precision, recall and f-measure. Also, number of training examples in our relevant and irrelevant classes

T3: `Most Urgent Need in #Nepal: Search and Rescue capacity; Medical teams, more supplies and tenting for hospitals and body bags`
T4: `Urgent need of analgesic,antibiotics, betadiene, swabs in kathmandu!! Call for help DIGIT #earthquake #Nepal`
T5: `Earthquake survivors in Nepal urgently need help. Make an emergency donation to WFP`

To test the performance of the system, we train machine learning classifiers. Specifically, using the returned results against our queries, we manually labeled which tweets are relevant and which are not. In this case, our classification task consists of two classes. We use Random Forest as our learning scheme. For evaluation, we use 10-fold cross validation technique and we report precision, recall, and f-measure scores to show the classification performance.

Table 1 shows the results of the classification task. We can observe that overall the system performs good. Other than the results of Q1, all performance indicators show results >0.80. We also observe a low precision in the case of Q1, whereas recall for Q1 is in the acceptable range.

User Studies

In order to perform further validation of the system's performance, we conducted user studies using 8 participants. All of our participants are either Master's students, PhD students or professionals. All of our participants use Twitter in their daily routine to get latest updates and to post tweets (on average 2 posts per day). Table 2 shows more details of our participants.

To start, we verbally described the motivation behind the system, how it works along with a short demo, and details of our Nepal earthquake dataset. We present the system to each participant and ask them to run two to three queries of their choice. Figure 2 shows the screenshot of the system interface. Each participant was instructed to mark the tweets that they think are relevant to their queries. The participants actively used the system and tried different queries. At the end, we asked them the following questions.

1. Overall, do you think the system fulfilled your information needs?
2. Overall, do you think the tweets returned by the system were mostly relevant to your query?
3. Did you observe an improvement in the returned results while using the system?
4. Do you think the number of tweets shown on a page is easy to go through?
5. Do you think system was difficult to use?
6. Will you consider this system to search information during a crisis event?

For the above six questions, we used the following scale for a participant's answer. (i) Strongly Agree, (ii) Agree, (iii) Neutral, (iv) Disagree, (v) Strongly Disagree

Finally, we ask each participant regarding how would he/she rate overall experience using the system. We used options: (i) Highly satisfactory, (ii) Satisfactory, (iii) Neutral, (iv) Unsatisfactory (v) Highly Unsatisfactory)

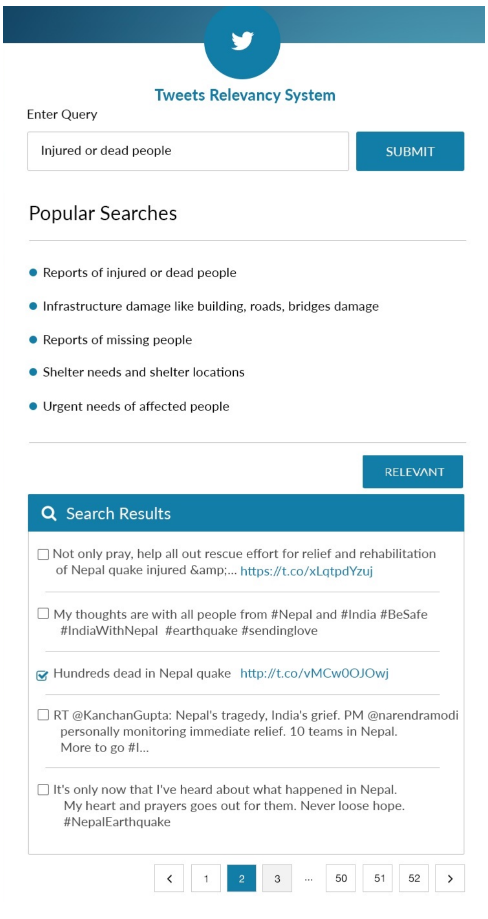

Figure 2. System used to show query results and to get users feedback

Table 2. Users demographic information

Users	Age	Gender	Education	Profession
U_1	30	Male	MSCS	Software Engineer
U_2	24	Male	BSSE	Entrepreneur
U_3	24	Female	MSIT	Software Engineer
U_4	26	Female	MSIT	Web Developer
U_5	35	Male	BSIT	SEO Expert
U_6	22	Male	BSCS	Android Developer
U_7	28	Female	PhD	Research Associate
U_8	30	Female	PhD	Student

Summary of Participants Feedback

Regarding the question 1, five out of eight participants agreed that the system indeed helped them get the information that they aimed for. The remaining three chose the neutral option. We also observe the same trend in the case of 2nd question (5 out of 8 agreed). However, whether the system returned better results during different iterations, 6 participants agreed that the system improved by time (the other 2 chose neutral option). Most of the participants did not seem to have any issues with the number of tweets shown on a page (question 4) and none of the participants showed concerns regarding system usage difficulties. The participants also showed interest to use the system in future. In regards to the last question, we observe a ratio of 1:5:2 for highly satisfactory, satisfactory, and neutral respectively.

Discussion

Retrieval of relevant documents from a live data stream is generally a challenging task. However, due to limited contextual information, processing and extracting relevant information from Twitter data stream is a non-trivial task. In this work, we have employed a state-of-the-art technique (word2vec) useful to increase contextual information given a reference word/term. Once trained on a large corpus, word2vec embeddings can be useful to retrieve most contextual relevant terms to a given input word. However, one of the limitations of this technique, which we also inherit, is the lack of large domain-specific corpus for the training of a good word2vec model. To overcome this issue, we aim to collect more disaster-related datasets form Twitter to see if it improves our current results.

Moreover, another challenge in the current approach is to achieve optimal query vectors in a timely manner. Since this process involves human input (i.e., marking of relevant results), there is an inherit complexity to determine the optimal number of results to be shown to the users. The aim here is to reach an optimal query vector in fewer iterations (i.e., using fewer human annotations). Furthermore, training supervised classifiers always demand a good amount of training data. In our current experiments, it is evident that the number of training examples are very limited. However, in future we aim to overcome this issue by employing unsupervised techniques to get highly similar messages to the ones annotated by the users.

RELATED WORK

Given the widespread use of social media platforms, especially Twitter, during crises opens new opportunities for people in the disaster zone and humanitarian organizations to fulfill their rapid information needs in many forms, such as textual summaries, ranked output, classified images (Rudra et al. 2016; Ofli et al. 2016). Twitter is not only useful to consume important information, but also to quickly and effectively disseminate relevant information (M. Mendoza and Castillo 2010). On average different types of users (Uddin et al. 2014) post 500 million tweets per day[3]. Extracting useful information from such huge data, especially in real-time, is a challenging task. Much of this online information contains irrelevant stuff. For this purpose, many techniques have been proposed to convert such big data into useful information (MacEachren et al. 2011; Cameron et al. 2012; Terpstra et al. 2012). However, many state-of-the-art approaches, for example, supervised classification techniques require event-specific training data (Imran, Elbassuoni, et al. 2013). Despite, recent studies have explored ways

[3] http://www.internetlivestats.com/twitter-statistics/

to utilize labeled data from past events (Imran, Mitra, and Srivastava 2016), however, the need of event-specific labeled data to train better classifier has not been fulfilled yet.

Recently, there have been advances in core technologies useful for natural language processing. For instances, word2vec is one of them and is being used for a number of application areas. Word2vec is a form of word vector representations. Vector space model (VSM) represents embedded words in a vector space where semantically similar words are placed near to each other. This has received a great deal of attention in the natural language processing and machine learning communities for their ability to model term similarity and other relationships (Diaz et al. 2016).

Refining a query either uses a fully automatic approach or it can be optimized by adding a human in the loop. Adding relevant words to the context and subtracting irrelevant ones will eventually help in getting the fully optimized query. Optimal query helps maximizing the similarity to relevant documents while on the other hand it will minimize the similarity to non-relevant documents. The relevance feedback algorithm is one of the most popular and widely applied learning methods from information retrieval (Joachims 1996)

There are many systems that use the above mentioned methods in order to find out the relevant information but none of them is actually applicable for information extraction in real-time situations without labeled data. There is no real time system which will take users relevance feedback during a crisis situation and learn a supervised model. To fill these gaps, we proposed a novel approach that uses word2vec, query expansion, query optimization techniques to automatically train supervised machine learning classifiers to fulfill users information needs.

CONCLUSION

Affected people and humanitarian organizations have quick information needs during time of disasters and emergency situations. Existing studies indicate the social media content posted during a crisis to fulfill peoples information needs. However, obtaining relevant information from social media platform such as Twitter is not a trivial task. Despite advances in natural language processing and supervised classification techniques, fast access to relevant information is still a challenging task. In this work, we propose a novel solution based on query expansion and relevance feedback mechanism to simultaneously retrieve relevant information and training of a supervised classifier. Initial results show the potential of the proposed approach. However, further investigation using datasets from other crises has to be done, which we consider as future work.

REFERENCES

Cameron, M. A., Power, R., Robinson, B., and Yin, J. (2012). “Emergency situation awareness from twitter for crisis management”. In: *Proceedings of the 21st International Conference on World Wide Web*. ACM, pp. 695–698.

Carpineto, C. and Romano, G. (2012). “A survey of automatic query expansion in information retrieval”. In: *ACM Computing Surveys (CSUR)* 44.1, p. 1.

Diaz, F., Mitra, B., and Craswell, N. (2016). “Query Expansion with Locally-Trained Word Embeddings”. In: *CoRR* abs/1605.07891.

Ghorab, M. R., Zhou, D., OConnor, A., and Wade, V. (2013). “Personalised information retrieval: survey and classification”. In: *User Modeling and User-Adapted Interaction* 23.4, pp. 381–443.

Hughes, A. L. and Palen, L. (2009). “Twitter adoption and use in mass convergence and emergency events”. In: *International Journal of Emergency Management* 6.3-4, pp. 248–260.

Imran, M., Castillo, C., Diaz, F., and Vieweg, S. (2015). “Processing social media messages in mass emergency: A survey”. In: *ACM Computing Surveys (CSUR)* 47.4, p. 67.

Imran, M., Castillo, C., Lucas, J., Meier, P., and Vieweg, S. (2014). “AIDR: Artificial intelligence for disaster response”. In: *Proceedings of the 23rd International Conference on World Wide Web*. ACM, pp. 159–162.

Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., and Meier, P. (2013). “Practical extraction of disaster-relevant information from social media”. In: *Proceedings of the 22nd International Conference on World Wide Web*. ACM, pp. 1021–1024.

Imran, M., Mitra, P., and Castillo, C. (2016). “Twitter as a Lifeline: Human-annotated Twitter Corpora for NLP of Crisis-related Messages”. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Portoroz, Slovenia: European Language Resources Association (ELRA).

Imran, M., Mitra, P., and Srivastava, J. (2016). "Cross-language domain adaptation for classifying crisis-related short messages". In: *In proceedings of ISCRAM*.

Joachims, T. (1996). *A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization.* Tech. rep. DTIC Document.

M. Mendoza, B. P. and Castillo, C. (2010). "Twitter under crisis: can we trust what we RT?" In: *Proceedings of the First Workshop on Social Media Analytics*.

MacEachren, A. M., Robinson, A. C., Jaiswal, A., Pezanowski, S., Savelyev, A., Blanford, J., and Mitra, P. (2011). "Geo-twitter analytics: Applications in crisis management". In: *25th International Cartographic Conference*, pp. 3–8.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*, pp. 3111–3119.

Ofli, F., Meier, P., Imran, M., Castillo, C., Tuia, D., Rey, N., Briant, J., Millet, P., Reinhard, F., Parkan, M., et al. (2016). "Combining human computing and machine learning to make sense of big (aerial) data for disaster response". In: *Big data* 4.1, pp. 47–59.

Rudra, K., Banerjee, S., Ganguly, N., Goyal, P., Imran, M., and Mitra, P. (2016). "Summarizing situational tweets in crisis scenario". In: *Proceedings of the 27th ACM Conference on Hypertext and Social Media*. ACM, pp. 137–147.

Starbird, K., Palen, L., Hughes, A. L., and Vieweg, S. (2010). "Chatter on the red: what hazards threat reveals about the social life of microblogged information". In: *2010 ACM conference on Computer supported cooperative work*, pp. 241–250.

Terpstra, T., De Vries, A., Stronkman, R., and Paradies, G. (2012). *Towards a realtime Twitter analysis during crises for operational crisis management*. Simon Fraser University.

Uddin, M. M., Imran, M., and Sajjad, H. (2014). "Understanding types of users on Twitter". In: *arXiv preprint arXiv:1406.1335*.