Classification of eyewitness tweets in emergency situations

Ioana Stefan
University Politehnica of Bucharest
313 Splaiul Independentei, Bucharest, Romania
ioana.stefan@stud.acs.upb.ro

Traian Rebedea
University Politehnica of Bucharest
313 Splaiul Independentei, Bucharest, Romania
traian.rebedea@cs.pub.ro

Doina Caragea
Kansas State University
2184 Engineering Hall, 1701D Platt St.
Manhattan, KS 66506, USA
dcaragea@ksu.edu

ABSTRACT
Social platforms such as Twitter offer important information about disasters in emergency situations. Unique information found on these platforms is provided by people directly involved. Presenting these data to rescue teams can make a significant difference in how the situation is managed and how resources are distributed. Identification of relevant tweets can be done with Machine Learning and Natural Language Processing techniques. Various supervised and unsupervised learning algorithms have been previously used for this problem, including diverse heuristics. The purpose of this project is to explore and compare several approaches, test variations of parameters, and filter input data in order to improve performance. Challenges posed by the class imbalance present in emergency situations and the language diversity on social media platforms are also discussed.

Author Keywords
Eyewitness tweets, Machine learning, Supervised learning, Classification

ACM Classification Keywords
Topic I.2.6 - Learning.

INTRODUCTION
Online social networks such as Twitter are very used nowadays and represent an important means to transmit information. One of the uses for Twitter and other social networks is to alert people in emergency situations, where speed is key. Diverse information can come from different sources, such as news channels or eyewitnesses. The latter are very important because their input is unique and hard to obtain. The information provided by them gives a better perspective of the situation and helps rescuers to organize their time and resources.

This project aims to classify tweets about emergency situations based on eyewitness and non-eyewitness sources.

RELATED WORK
Morstatter et al. [1] illustrated the fact that most tweets are not geotagged, a feature that would have been useful in solving the eyewitness identification problem. As a result, they proposed a model based on language and linguistic patterns. The characteristics taken into consideration are: time, words differences and linguistic features. Linguistic features include the counting of unigrams and bigrams and also crisis-sensitive features (specific words found to be characteristic of an emergency situation). The crisis-sensitive features are shown to be similar for different emergency situations, an observation that we also used in this work.

Purohit et al. [2] described a system for the discovery and ranking of requests that can be serviceable. An eyewitness will transmit unique information, most of the time either as a request for help or as a specific disaster place where an intervention team can action. A request is considered to be serviceable if it is related to a certain topic and contains the necessary details, such as time, place, and context. These kinds of details were also added to the set of keywords used in this paper. The final model proposed by Purohit et al. [2] is the combination of two other models: the Qualitative serviceability model, which contains all the necessary information, and the Quantitative serviceability model, which assigns a score for each characteristic of a request, similar to an attention mechanism.

An approach for detecting eyewitness tweets is to identify specific characteristics for the posts of interest. Zahra et al. [3] realized a classification of tweets and their characteristics. Their classification includes direct eyewitness, indirect eyewitness and non-eyewitness. The most important characteristics for a direct eyewitness post consist of first-person pronouns and adjectives, time indicating words, impact, short length, personalized location markers, intensity indication, small details, and perceptual senses. Most of these characteristics can be found in our set of keywords too.

Li et al. [4] described a domain adaptation approach based on Naive Bayes classifier intended for Twitter posts. This approach takes into consideration the frequency of each word occurrence in different documents/tweets. An

algorithm used for our classification of eyewitness tweets was inspired by their work.

Anomaly detection is a common problem where the goal is to identify sparse types of information through large datasets. Du et al. [5] worked on a deep learning algorithm solution for this problem. Two models are presented in their paper: a traditional N-gram model and a model based on stacked long short-term memory (LSTM) networks. Both approaches depend on a unique log key constructed for each input. The inherently class imbalanced data available in the disaster domain inspired us to adapt an anomaly detection approach to our classification problem.

Ionescu et al. [6] have approached the detection of abnormal events by balancing the input data. To do this, new classes were created and the examples were classified accordingly. The new classes were created based on an unsupervised learning classification on the input data. The purpose behind these new classes was to make a deeper separation of the data and to increase the likelihood of an example to be part of any class equally. Similarly to the previous described approach, this idea was also implemented for our classification problem in order to address the class imbalance problem.

Neppalli et al. [7] compared Naive Bayes classifiers and Deep Neural Networks (DNN) approaches. The context of this comparison is the identification of informative tweets from disaster situations. Their paper shows that neural networks have better overall results. Similar to their paper, we perform a comparison of these algorithms in the evaluation section, obtaining similar results.

In this project, we use the algorithms presented in the aforementioned papers, and experiment with a variety of approaches. The eyewitness characteristics identified are taken into consideration by filtering the input data and creating a set of keywords to be considered in one of the input representations.

ALGORITHMS AND HEURISTICS

Different supervised learning algorithms were used for the classification of eyewitness tweets: Naive Bayes, Logistic Regression, and Neural Networks. The latter has been tested with multiple types of layers and embeddings.

The data corresponding to emergency situations is almost always class imbalanced, since the number of eyewitness tweets is significantly smaller than the number of tweets posted by organizations and news channels that cover an emergency. An approach of anomaly detection has been adapted for the eyewitness classification. Another approach adapted to this problem in order to address the class imbalance problem was the creation of new classes.

Two types of inputs extracted from tweets have been provided to these algorithms:

- Each example is an array of features, which were selected based on eyewitness and non-eyewitness characteristics, such as personal pronouns, senses, and specific keywords used in emergency situations (Table 1). These features were represented in the array in two ways: binary representation and counting representation.
- Each example is a paddled array of integer embeddings, pre-trained or trained in place.

Type of words	Examples
Pronouns	I, me, ours
Senses	see, feel, hear
Demonstratives	this
Time-specific words	tomorrow, tonight, night, day
Location-specific words	here, there, north, east
Actions	talk, show, view, work, lose
Feelings	sad, condolences, sympathy
Event characteristic words	victim, donation, news, coverage, official, wounded, aid, money, blood

Table 1. Types of features selected to represent a tweet.

Naive Bayes

The first type of input was used for this algorithm. The testing was done with Naive Bayes Bernoulli and Naive Bayes Multinomial.

Logistic Regression

The first data representation was used for this algorithm. Different numbers of iterations have been used to identify the approximate point of convergence.

A linear separation of the two considered classes was done based on the attributes of each example. The Logistic Regression model uses the sigmoid function. The result of the sigmoid function is compared with a threshold of 0.5: a result equal or above the threshold is considered an eyewitness tweet.

The model uses gradient descent to train the model, with the cost function presented in Equation 1.

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)} log h_\theta(x^{(i)}) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

Equation 1. Cost function for the linear regression algorithm.

In Equation 1, m represents the number of training examples, $y^{(i)}$ is the actual class of example $x^{(i)}$ and $h_\theta(x^{(i)})$ is the predicted class of example $x^{(i)}$.

Neural Networks

We have used networks with two and three hidden layers, of different types and numbers of cells. Densely-connected and LSTM layers have been used. The number of cells was varied between 10 and 150. Training the neural networks used the Adam optimizer and Binary Cross-entropy as the loss function.

The neural networks that include densely-connected cells use the first input representation and the pre-trained embeddings representation, while the ones that include LSTM use the representation with embeddings trained in place. In addition, the Bag of Words model is used in one neural network to construct the embeddings.

Anomaly detection

The first data representation is used in this case.

Given the case of highly imbalanced data, the eyewitness tweets were approached as an anomaly detection problem. The algorithm used here is based on the Gaussian distribution. Each different feature given as input influences the final prediction. The Gaussian distribution is calculated for each of these features. A Gaussian distribution is represented by the following parameters: the mean and the variance. These parameters are described in the first two formulas in Equation 2.

The final decision for an example is given by the comparison with a given constant limit epsilon. A probability is calculated for each example and if this probability is lower than the limit, then that example is considered an anomaly. The probability for a given example is calculated based on the final formula in Equation 2.

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$\sigma^2 = \frac{1}{m}\sum_{i=1}^{m} (x^{(i)} - \mu)^2$$

$$p(x) = \prod_{j=1}^{n} p(x_j; \mu_j; {\sigma_j}^2) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_j} exp(-\frac{(x_j-\mu_j)^2}{2{\sigma_j}^2})$$

Equations 2. Anomaly detection related formulas.

Multi-class problem

The first data representation was used for this approach.

The idea behind this problem was to create multiple new classes, such that at the end the classes will be approximately balanced. The problem that needs to be solved is a multi-class classification problem. This problem was further solved with a neural network with one hidden layer formed of densely-connected cells. The Adam optimizer and the Categorical Cross-entropy loss function were used.

There are two ways in which the new classes are created:

- By manual labeling of the examples in categories other than eyewitness / non-eyewitness. The categories used for this approach are: affected individuals; infrastructure and utility damage; injured or dead people; missing or found people; rescue; volunteering or donation effort; vehicle damage; other relevant information.
- By classifying the examples with an unsupervised learning algorithm. The K-Means clustering algorithm was used, with different numbers of centroids. A neural network was applied on top of the information learned in the clustering phase. At the end, the target data prediction was compared with the actual labels, created in the first approach.

Attention mechanism

A Gated Recurrent Unit (GRU) cell is a simplified version of an LSTM cell, which includes two gates, compared to an LSTM cell, which includes three gates. A GRU layer was used in this approach in order to reduce complexity.

A Bahdanau attention mechanism [8] has also been taken into consideration. This model contains Encoder and Decoder layers, formed of embeddings trained in place and GRU layers.

The score in Bahdanau Attention is computed and a softmax function is applied on top of it, in order to obtain the attention weights. The context vector is obtained as the dot product between the attention weights and the output of the Encoder layer.

$$score = {v_a}^T tanh(W_1 h_t + W_2 h_s)$$

$$context = \sum_s \alpha_{ts} h_t$$

Equation 3. Attention mechanism related formulas

IMPLEMENTATION

The project was implemented in Python.

The approach that uses anomaly detection has been implemented based on the formulas described previously.

The sklearn library [9] has been used for the implementation of the Naive Bayes and Logistic Regression classifiers. This library has also been used for the unsupervised classification of data in the multi-class problem.

All the neural networks implementations use the Tensorflow framework [10] and the Keras Python library [11]. The embeddings used have been constructed using the Embedding layer offered by Keras, along with the Bag of Words model, or pre-trained embeddings offered by GloVe [12].

Datasets

The CrisisMMD dataset [13] was used for evaluation. This dataset does not include eyewitness information and has been partially labeled manually with this information as part of this project. Around 1000 tweets were labeled. The class distribution of the datasets used in the experiments for two and eight classes, respectively, can be found in Table 2.

It has been observed that additional parsing of the input data brings considerable improvements to the results. This parsing extracts only the examples that do not include links in the text. This filter is based on the idea that eyewitnesses do not have time to include links into their posts or do not want to do this, given that their information is already valuable by itself.

Some examples of eyewitness and non-eyewitness tweets can be found in Table 3.

Dataset / Classes	California Fires	Iran Iraq Earthquake	Hurricane Harvey
Two	69% - 31%	86.4% - 13.6%	76% - 24%
Eight	7.9% - 6.3% - 9.5% - 0% - 6.3% - 0% - 39% - 31%	7.4% - 4.4% - 15.7% - 0% - 12.2% - 0% - 46.7% - 13.6%	2.8% - 8% - 2% - 0% - 23% - 0.4% - 39.8% - 24%

Table 2: Class distribution for CrisisMMD dataset. Each percentage represents the number of examples that correspond to a class from the number of total examples. The last percentage in each entry corresponds to the eyewitness class.

Datasets	
California Fires	
Non-eyewitness	**Eyewitness**
Neighborhoods swallowed up by flames as toll rises in California wildfires CLICK BELOW FOR FULL STORY... …	A hill is being silhouetted by flames during the Nuns Fire in Kenwood, California
Eastern #SantaRosa is still under #redflag warning #evacuation.. California #wildfire	Emergency medical supplies delivered tonight to Sonoma Public Health for evacuees of Northern California #Wildfires
Hurricane Harvey	
Non-eyewitness	**Eyewitness**
RT @JMilesKHOU: Rockport, TX damage via @StormVisuals. Feeling for those folks #HurricaneHarvey #khou11	RT @stephentpaulsen: My street in SE #Houston is now a river. That light is from lightning; it's 10pm #Harvey
Hurricane #Harvey wind map Sat midnight	#Harvey #Pasadena This is an update on the water level.
Iran-Iraq Earthquake	

Non-eyewitness	Eyewitness
Please accept my heartfelt sympathy on the passing of your dears #earthquake#kermanshah#Iran	RT @imranbird: #Earthquake in #kuwait. Everyone is standing outside.
Tragedy.....earthquake in iran	Trying to find info re U.S. troops & personnel operating in/around Iran-Iraq earthquake region - nothing so far.

Table 3: Examples of eyewitness and non-eyewitness tweets found in the CrisisMMD dataset.

Algorithm / Dataset	California Fires	Iran-Iraq Earthquake	Hurrican e Harvey
Naive Bayes Bernoulli (sklearn)	0.28	0.34	0.28
Naive Bayes Multinomial (sklearn)	0.28	0.64	0.33
Logistic Regression	0.28	0.14	0.25
Neural Network with one hidden Densely-connected layer - softmax output	0.65	0.57	0.61
Neural Network with pretrained embeddings - Bag of Words - and one LSTM hidden layer	0.76	1.0	0.82
Neural Network with pre-trained embeddings - GloVe	0.79	1.0	0.81
Neural Network with pretrained embeddings - GloVe - and one LSTM hidden layer	0.67	0.85	0.32
Anomaly detection	0.68	0.44	0.61
K-Means for labeling data and Neural Network with one Densely-connected hidden layer	0.8	0.94	0.815
Attention mechanism	0.79	0.77	0.85
Attention mechanism with pre-trained embeddings - GloVe	0.77	0.77	0.89

Table 4: CrisisMMD dataset results (Mean Average Precision Score) for binary classes.

Algorithm / Dataset	California Fires	Iran Iraq Earthquake	Hurrican e Harvey
Neural Network with manually labeled data	0.42	0.57	0.66
K-Means for labeling data, and Neural Network with one Densely-connected hidden layer	0.37	0.26	0.3

Table 5: CrisisMMD dataset results (Mean Average Precision Score) for multiple classes.

RESULTS

All the algorithms have been tested on the CrisisMMD datasets corresponding to the following disasters: California fires, Iran-Iraq Earthquake, Hurricane Harvey. Tables 4 and 5 illustrate the differences between the algorithms for the identification of eyewitness tweets for two and eight classes, respectively.

Important differences have been observed after filtering the input data, for multiple algorithms.

The class imbalance has proven to be a challenge because it creates significant differences between class weights. Even though this project has tried to solve the issue by restructuring the problem as an Anomaly Detection problem and a Multiclass problem, the overall best performances have been obtained using Neural Networks. This algorithm has encountered differences based on the type of layers and inputs used, proving that tuning of the parameters and structure is needed in order to find a good balance between precision and recall.

The results presented in Tables 4 and 5 represent the best results obtained by varying the parameters of the algorithms and by keeping the best model out of a set of models for each algorithm.

While the elimination of tweets that contain links has been beneficial for the final results, this method reduces the size of a dataset to a small number of examples, which are prone to overfitting.

Examples

Some examples of correct and incorrect classification realized with one of the best algorithms implemented (Neural network with pre-trained embeddings) are presented in what follows.

Correct classification:

On the road to our live shot location this afternoon: Some of what the California wildfires left behind. - eyewitness

At least 70,000 people still unable to go home tonight as California wildfires continue raging. – non-eyewitness

Pics from Iran 204 person killed & 1600 injured by #earthquake yesterday #زلزال_ایران – non-eyewitness

Heavy debris removal vital to recovery. This was welcome sight near Fiesta on 59N, @SylvesterTurner. – eyewitness

Tree down in NW HIlls#HurricaneHarvey@statesman - eyewitness

Incorrect classification:

Trump unveils plan to fight California wildfires..... – non-eyewitness

#Ateam! #hcphtx at NRG everyday addressing medical/#publichealth issues post #Harvey! #recovery is in full – eyewitness

A big difference can be observed here when comparing the model with a manual classification. The correctly classified tweets present precise information and have a text that is easy to parse. While a person would have been able to classify the last two tweets, the model fails to do so because of possible different reasons: the precise information and the present tense might be a reason for considering the first of the two an eyewitness tweet; the parsing done for this project did not include hashtags, which makes the second tweet to have too little information for a correct prediction.

CONCLUSION

The detection of eyewitness tweets has proven to be difficult because of the multiple features that can influence an eyewitness post and because of the sparse data available. Taking into consideration that people talk differently, as well as the elimination of connection words in short texts, a perfect model of one's speech is hard to be achieved. In addition, the small size of the datasets used is also a challenge. Once bigger more balanced datasets will be acquired, the results are expected to improve considerably.

This project is different compared to the related work in the same field as multiple algorithms and approaches have been tested. Some of these algorithms have not been previously used in relation to the classification for eyewitness tweets.

As future work, a deeper parameter tuning might bring better results. Testing with more approaches is also of interest, especially semi-supervised, domain adaptation and unsupervised learning algorithms.

ACKNOWLEDGEMENTS

We thank the National Science Foundation and Amazon Web Services for support from grant IIS-1741345, which partially enabled the research in this study.

REFERENCES

1. Morstatter, F., Lubold, N., Pon-Barry, H., Pfeffer, J., & Liu, H. (2014). Finding eyewitness tweets during crises. arXiv preprint arXiv:1403.1773.
2. Purohit, H., Castillo, C., Imran, M., & Pandev, R. (2018, August). Social-eoc: Serviceability model to rank social media requests for emergency operation centers. In 2018 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 119-126). IEEE.
3. Zahra, K., Imran, M., Ostermann, F. O., Boersma, K., & Tomaszewski, B. (2018). Understanding eyewitness reports on Twitter during disasters. In ISCRAM 2018 Conference Proceedings (pp. 687-695).
4. Li, H., Caragea, D., Caragea, C., & Herndon, N. (2018). Disaster response aided by tweet classification with a

domain adaptation approach. Journal of Contingencies and Crisis Management, 26(1), 16-27.

5. Du, M., Li, F., Zheng, G., & Srikumar, V. (2017, October). Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1285-1298). ACM.
6. Ionescu, R. T., Khan, F. S., Georgescu, M. I., & Shao, L. (2018). Object-centric Auto-encoders and Dummy Anomalies for Abnormal Event Detection in Video. arXiv preprint arXiv:1812.04960.
7. Neppalli, V. K., Caragea, C., & Caragea, D. (2018). Deep Neural Networks versus Naive Bayes Classifiers for Identifying Informative Tweets during Disasters. In ISCRAM.
8. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
9. Sklearn library. https://scikit-learn.org/stable/, last accessed at 13 July 2019.
10. Tensorflow framework. https://www.tensorflow.org/, last accessed at 15 July 2019.
11. Keras library. https://keras.io, last accessed at 15 July 2019.
12. GloVe pretrained embeddings. https://nlp.stanford.edu/projects/glove/.
13. Alam, F., Ofli, F., & Imran, M. (2018, June). Crisismmd: Multimodal twitter datasets from natural disasters. In Twelfth International AAAI Conference on Web and Social Media.