NATURAL LANGUAGE PROCESSING WITH DISASTER TWEETS

BITS Pilani Hyderabad Campus

CS F429 Natural Language Processing Project

Manasa Reddy Bollavaram
f20180774@hyderabad.bits-pilani.ac.in
Yashika Chopra
f20180894@hyderabad.bits-pilani.ac.in
Arasanapalai Srikanth Navya Sri
f20180134@hyderabad.bits-pilani.ac.in
Adivikolanu Sri Naga Rishika
f20180730@hyderabad.bits-pilani.ac.in

Abstract

This paper focuses on the classification of disaster related tweets into real and not real categories. These days, social media is playing a crucial role in providing information before, during, and after certain unforeseen circumstances such as natural disasters. Through social media both victims and officials can put their problems as well as the solutions and suggestions at the same place in real time. One of the major social media platforms used for the collection of disaster related information is twitter. The tweets are usually written by the people in an area where the disaster took place. Tweets may contain information of multiple purposes which include seeking help, expressing grief, warning others about the disaster etc. Some tweets are also written by the general public to bring awareness among others. The key component in Natural language processing involves understanding the events described in the text. So, identifying the tweets which actually talk about a real disaster is very important and we have tried multiple word representations, topic modeling techniques and text classification techniques for this purpose. We have used certain evaluation metrics like accuracy, precision, recall and f1-score to test the models.

KEYWORDS: Natural Language Processing, Twitter, Disaster Identification, Social Media, classification.

1.INTRODUCTION

When a disaster occurs, all the disaster relief organizations, NGOs need to get the real time information about the help required by the victims. The disaster affected area requires both cautionary and disciplinary measures. These days because of the omnipresence of smart phones, laptops, tablets everyone is using social media and this information can be easily provided from the data through different social media platforms. And spreading information via Social media can be done very quickly. Social media is a platform to express one's view in real time. This real time nature of social media is the key factor that can help in disaster management. And in recent times, among all the other social platforms, twitter has become an important communication channel particularly during emergency times. Twitter plays an important role in informing people, acquiring the status information of affected people, and also gathering information on various rescue activities. Disasters can be classified into 2 categories namely natural disasters like tsunamis, earth quakes, floods and man-made disasters like terrorist attacks, food contamination etc. Twitter can become a useful source during both these disaster types. Because of this, many people and many agencies are trying to monitor and analyze the twitter data. Particularly, many news agencies and disaster relief organizations are trying to analyze tweets in real time to identify the occurrence of disasters from tweets, so that they react immediately and help the needy . This would help millions of common people by preventing the danger from impending disasters. If people could be informed regarding the occurrence of disasters at a particular location in real-time then they can take quick actions. Government agencies can execute evacuation before the situation goes beyond control.

But the twitter data is usually very unstructured because the data is not written in any specific format. So analyzing the twitter data is a major challenge. For this purpose Natural Language Processing (NLP) models are used. The tweets are tokenized into words as a part of preprocessing. Then analysis is performed on this preprocessed dataset. So far the research work that has been done in the field of natural language processing with respect to disaster tweets has primarily focused on classifying tweets which are already known to refer to a real disaster. But in some cases it is not always clear whether a person's words are actually about a real disaster or they are being metaphorical. Tweets can sometimes be misleading. Also while classifying disaster tweets into real and not real categories, most of the work has been done using supervised classification techniques. Unsupervised techniques have not been explored much. And even during the classification, sentiment analysis has not been used. Sentiment analysis is the process of extracting subjective qualities from the text such as emotion or

attitude. The aim of Sentiment analysis is to find the sentiment of a tweet (positive or negative). Over the years, with the advancement of technology and Natural Language Processing methodologies, the process of text and sentiment analysis has become much easier than earlier.

The data that this paper analyzes consists of 7613 tweets along with the corresponding tweet-id , location ,tweet keyword and the target variable. The target variable is a class to which the tweet belongs. Class 1 indicating a disaster related tweet and Class 0 indicating Not Disaster related tweet. We have used this dataset to train and test the models. The dataset is cleaned and tokenized and then it is divided into training and testing sets for checking the accuracy of multiple models. In this project, Word2Vec, TF-IDF, Bertweet, Glove and FastText word embeddings are used. LDA (with BERT, Word2Vec) and LSA (with TF-IDF) are modeled which come under unsupervised category. Different text classification (supervised) techniques like Fine tuning BERT, Multilayer Perceptron (MLP) classifier, Long short-term memory (LSTM) classifier, Extreme Gradient Boosting (XGBoost), Support Vector Machine (SVM) are implemented. For evaluating all these models different metrics like accuracy, precision, f1-score and recall are used. The programming language used is python. Google Colab is used for coding.
If a person tweets a message which is about an emergency or a forthcoming disaster and if this can be recognized immediately by our NLP models, we would be able to react quicker than normal which would help save millions of lives. This is the purpose of the project. So the main objective of this project is to identify if a specific tweet is talking about a real disaster or not and finding out the best NLP model for this purpose.

2. OBJECTIVE

In this project, we tried to achieve the following objectives:

1. Compare different word embeddings with topic classification/modelling techniques
2. See how supervised and unsupervised techniques work for determining whether a tweet is about a real disaster or not.
3. Exploring recently developed models like BerTweet and see if they yield better results than the existing ones.

3. RELATED WORK

The authors Singh, J.P., Dwivedi, Y.K., Rana, N.P. et al explained about classifying tweets of flood related disasters into low priority and high priority and then finding the location of the disaster for high priority tweets. For classifying, 3 techniques were tried - SVM, random forest and gradient boosting. SVM performed poorer than the other 2. For finding the location, the 3 methods used were - address given in the tweet, geo tagging and Markov chain. They got a classification accuracy and location accuracy of 81% both. However, these models were trained specifically for flood related disasters and classifying those into high priority or low priority. (Singh, J.P., Dwivedi, Y.K., Rana, N.P., 2019) [1]. Kevin Stowe, Michael Paul, Martha Palmer, Leysia Palen, Ken Anderson focused on the 2012 Hurricane Sandy event, and presented classification methods for categorizing tweets relevant to the hurricane into fine-grained categories such as preparation and evacuation. Count of unigrams with full feature set as well as truncated set using standard selection technique was used. Feature selection proved to be substantially better than using all unigrams. 3 classification models were tried - SVM, MaxEnt and Naive Bayes of which SVM gave better results with feature selection. Only supervised classification techniques were used here. (Kevin Stowe, Michael Paul, Martha Palmer, Leysia Palen, Ken Anderson) [2]. Authors Chandan Mannem, Mahalavanya Sriram, Aparajitha Sriram distinguished if a tweet talks about a real disaster or not. They tried out a variety of different models, and found that the BERT model with a K-fold cross validation worked best. They got a classification accuracy of 79%. But, this was done without taking into consideration the topic of sentiment analysis. (Chandan Mannem, Mahalavanya Sriram, Aparajitha Sriram) [3]. The authors Md Johirul Islam, Hamid Bagheri worked on developing a predictive model for sentiment analysis of twitter. Textblob python library was used for text processing and NLTK for Natural language Processing. They classified tweets based on movies, politics, fashion, fake news, justice and humanity into 3 categories of tweets – positive, neutral and negative. (Bagheri

H , Islam Md J, 2017) [4]. In the paper written by authors Shriya Goswami , Debaditya Raychaudhuri, the basic objective was to classify the tweets into 2 classes namely Disaster related tweets and non-disaster tweets. The classification algorithm used was Decision Tree Classification Algorithm. Accuracy checking was done by generating a Confusion Matrix. Accuracy measurement was also done by calculating the AUC score. R programming language, which is one of the most popular data analytics, was used for implementing the algorithm. (Authors Shriya Goswami , Debaditya Raychaudhuri, 2020) [5]. Guoqin Ma in his research paper applied deep learning techniques to address Tweets classification problems in the field of disaster management. The labels of Tweets usually reflect several types of disaster related information, which may have multiple uses particularly during emergency times. He has used the standard BERT architecture for classification and several other customized BERT architectures were trained for comparing the baseline bidirectional LSTM with pretrained Glove Twitter embeddings. In his research work, BERT and BERT-based LSTM gave the best results, outperforming the baseline model by 3.29% on average in terms of F-1 score respectively. Bidirectional LSTM with GloVe Twitter embeddings was used as a baseline.(Guoqin Ma) [6]. The main goals of authors Himanshu Shekhar, Shankar Setty in their paper were Firstly,demonstrating geographical distribution of certain selected natural disasters within a given time period only by using the information from tweets.Secondly, providing continent-wise occurrence frequency of selected natural disasters. Thirdly, analysing people's sentiment during a disaster by applying sentiment-analysis on the tweet content. (Himanshu Shekhar, Shankar Setty, 2015) [7]. Authors S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen analyzed microblog posts generated during two concurrent emergency events in North America throughTwitter, which is a popular microblogging service. They focused on communications broadcast by people who were "on the ground" during the Oklahoma Grassfires of April 2009 and the Red River Floods that occurred in March and April 2009, and identified information that they felt would contribute in enhancing situational awareness. This work is aimed at creating further steps for extracting useful, relevant information during emergency times using information extraction (IE) techniques. (S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, 2010) [8]. The authors Ali Hasan, Sana Moin, Ahmad Karim, and Shahaboddin Shamshirband had collected tweets pertaining to certain political topics and hashtags originating in Pakistan and performed sentiment analysis on the tweets using Naive Bayes and SVM classifier. They used Textblob , SentiWordNet and W-WSD for text analysis and did a comparison after classifying the tweets into positive , neutral and negative categories. (Hasan A , Moin S, Karim A , Shamshirband, 2018) [9] The authors Emmanouil K. Ikonomakis , S.Kotsiantis and V. Tampakas worked in the

field of text mining , feature selection and feature transformation for text data analysis.The PCA and Document term matrix was harped upon. They used machine learning algorithms KNN and SVM to show how accuracy, recall and precision can be calculated. (Ikonomakis Emmanouil K , Kotsiantis S , Tampakas V , 2005) [10].

4. APPROACH

4.1 Data Collection:

The data which is used in this research paper is taken from the KAGGLE portal for data analytics and machine learning. It is a competition dataset related to the disaster tweets classification project. The training dataset csv file from the portal has been taken. And we have divided the entire data into training and testing sets with appropriate ratios for different models. The dataset consists of 7613 tuples (rows) which represent 7613 tweets. There are 5 attributes (columns) in the dataset which are Tweet Id , Text , Location , Keyword and the Target variable. Target variable is the response or the output. It is the dependent variable which is classified into 2 classes – 1 (Tweets related to disasters) or 0 (Tweets not related to disasters). For the test dataset we need to predict the value of the target variable.

4.2 Text Preprocessing:

Tweets contain different types of noise and redundancies, such as emoticons, user mentions, hash tags, Internet links etc. A proper data preprocessing is needed in order to convert these tweets into a meaningful sentence.

In this project, following preprocessing steps have been performed on the dataset :

1. Removed the location column as 33% of the location data contained null values
2. Removed the keyword column as we were not using it
3. Converted the entire tweet text to lower case to avoid increase in the size of the vocabulary for words written in different cases
4. Removed all the duplicate tweets
5. Removed internet links starting with http/s, www and hashtags and user mentions. The Internet links need to be deleted from the tweets, as we were concentrating on the textual content only.
6. Removed digits from the tweets
7. Removed punctuations

8. Removed stop words for tf-idf. Stop words are removed to eliminate low-level information from the text in order to give more focus to the important information.
9. Shuffled the rows in the data frame
10. Tokenization was performed for models using Word2Vec, LDA2Vec, models using FastText, models using GloVe. Tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words.

4.3 Word Embeddings/Vectorization:

We have used the following word embedding or vectorization techniques:

1. Word2Vec
2. FastText
3. GloVe
4. TF-IDF
5. BERTweet

LDA (with BERT, Word2Vec) and LSA (with TF-IDF) are modeled which come under unsupervised category. Different text classification (supervised) techniques like Fine tuning BERT, Multilayer Perceptron (MLP) classifier, Long short-term memory (LSTM) classifier, Extreme Gradient Boosting (XGBoost), Support Vector Machine (SVM) are implemented.

4.4 Topic Modeling (Unsupervised):

We have used the following topic modelling techniques.

1. Latent Dirichlet Allocation (LDA)
2. Latent Semantic Analysis (LSA)

4.5 Topic/ Text classification(Supervised):

We have used the following topic classification techniques.

1. Multilayer Perceptron (MLP)
2. Long Short-Term Memory (LSTM)
3. Extreme Gradient Boosting (XGBoost)
4. Support Vector Machine (SVM)
5. Logistic Regression

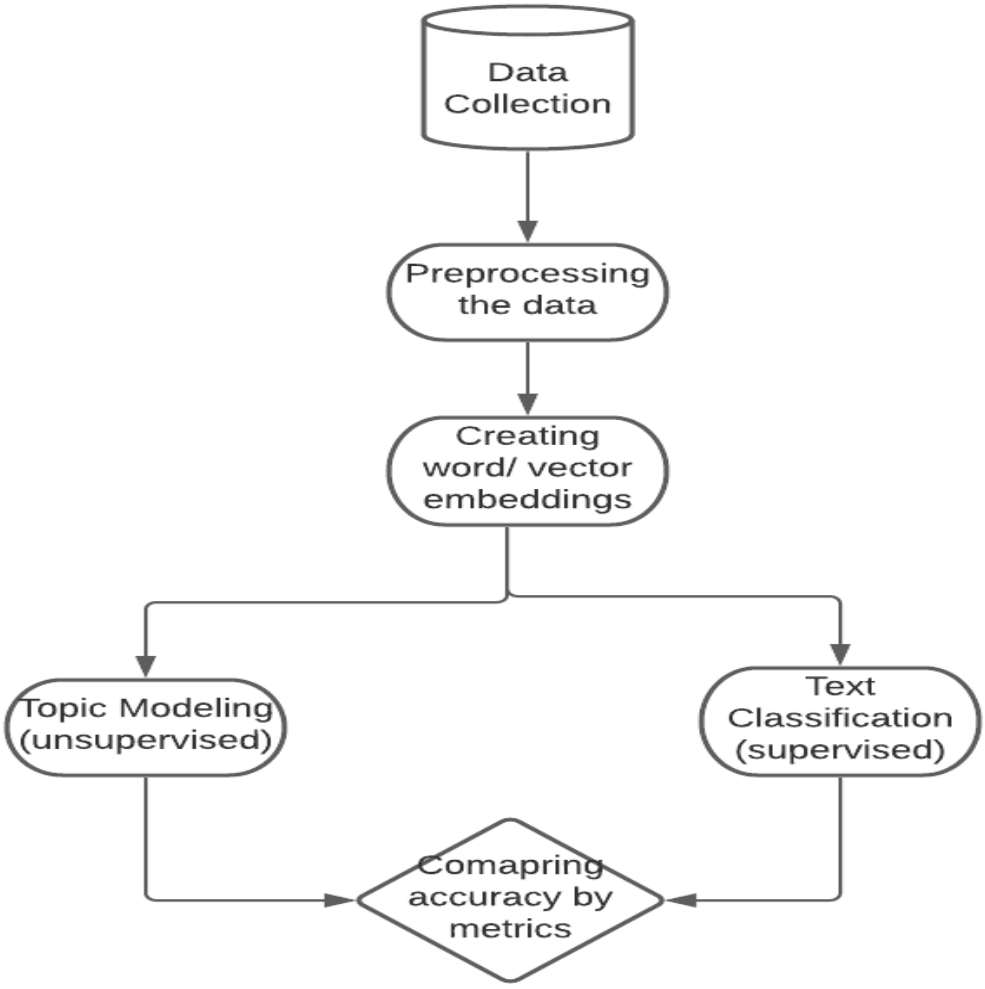
We have used the above mentioned word embeddings along with the topic modelling/classification techniques as follows:

1. Word2Vec embeddings with
    a. MLP
    b. LSTM
    c. XGBoost
    d. SVM
2. FastText embeddings with
    a. MLP
    b. LSTM
    c. XGBoost
    d. SVM
3. GloVe embeddings with
    a. MLP
    b. LSTM
    c. XGBoost
    d. SVM
4. TF-IDF vectors with MLP
5. LSA with Logistic Regression
6. LDA with Word2Vec (LDA2Vec)
7. BERTweet with MLP

4.6 Compare using metrics:

Accuracy, Precision, Recall and F1-score are used for comparing all the models.

METHODOLOGY FLOWCHART



5. EXPERIMENTS:

5.1 Dataset:

The data has been taken from the KAGGLE portal for data analytics and machine learning. We have divided the data into training and testing sets. The dataset contains 7613 tuples (rows) which represent 7613 tweets. There are 5 attributes (columns) in the dataset namely Tweet Id, Text/ Tweet, Location of the tweet, Keyword and the Target variable. The Target variable is a binary variable which represents either 0 or 1. 1 indicates a real disaster and 0 indicates that the tweet is not about a real disaster. The entire data is divided into training and testing sets with appropriate ratios for different models.

Each row of the data contains the following :
id - a unique identifier for each tweet
text - the text of the tweet
location - the location the tweet was sent from (may be blank)
keyword - a particular keyword from the tweet (may be blank)
target - this denotes whether a tweet is about a real disaster (if equal to 1) or not (if equal to 0)

95	ablaze	San Francisco	@ablaze what time does your talk go until? I don't know if I can make it due to work.	0
96	accident	CLVLND	'I can't have kids cuz I got in a bicycle accident & split my testicles. it's impossible for me ...	0
97	accident	Nashville, TN	Accident on I-24 W #NashvilleTraffic. Traffic moving 8m slower than usual.	1

Fig: Snapshot of data

5.2 Evaluation method/Metrics

The following metrics were used:

1. F1-score: F1-score was used for comparing the following models.
    a. Word2Vec with MLP
    b. Word2Vec with LSTM
    c. Word2Vec with SVM
    d. FastText with MLP
    e. FastText with LSTM
    f. FastText with SVM
    g. Glove with MLP
    h. Glove with LSTM
    i. Glove with SVM
    j. TF-IDF with MLP
    k. LSA with Logistic Regression
2. Cross-Validation score: Cross-validation score was used to evaluate the following models.
    a. Word2Vec with XGBoost
    b. FastText with XGBoost
    c. Glove with XGBoost
3. Coherence score: Coherence score was used to evaluate the LDA2Vec model.
4. Accuracy: Accuracy was used to evaluate the BERTweet model.

5.3 Experimental setup

The hyperparameters are as follows:

MODEL	HYPERPARAMETERS
Word2Vec	Sg = 0 (=> CBOW) Window size = 5 Size (Dimensions) = 100 Min_count = 1 Workers = 4
-with MLP	Number of layers = 4 -Embedding layer: number of neurons = 100 -Flatten -Dense layer 1. number of neurons = 500 2. activation function: relu -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy

-with LSTM	Number of layers = 3 -Embedding layer: number of neurons = 100 -LSTM number of neurons = 128 -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy
-with SVM	Gamma = auto
FastText	Window size = 3 Size (Dimensions) = 100 Min_count = 1 Workers = 4
-with MLP	Number of layers = 4 -Embedding layer: number of neurons = 100 -Flatten -Dense layer 1. number of neurons = 500 2. activation function: relu -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy

	Optimizer: Adam Metric: accuracy
-with LSTM	Number of layers = 3 -Embedding layer: number of neurons = 100 -LSTM number of neurons = 128 -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy
-with SVM	Gamma = auto
GloVe	

-with MLP	Number of layers = 4 -Embedding layer: number of neurons = 100 -Flatten -Dense layer 1. number of neurons = 500 2. activation function: relu -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy
-with LSTM	Number of layers = 3 -Embedding layer: number of neurons = 100 -LSTM number of neurons = 128 -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy
-with XGBoost	Gamma = auto
TF-IDF	

-with MLP	Number of layers = 2 -Dense layer 1. number of neurons = 500 2. activation function: relu -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy
BERTweet	Number of layers = 2 -Dense layer 1. number of neurons = 512 2. activation function: relu -Dense layer 1. number of neurons = 2 2. activation function: softmax Loss function: Binary Cross entropy Optimizer: Adam Metric: accuracy

For the neural networks-

- We experimented with 1 and 2 hidden layers, but only 1 hidden layer was sufficient.
- For the neurons, we experimented with 2/3rd the number of neurons in the previous layer with a difference of around 10 neurons.
- For the activation function, we tried relu and tanh but relu gave better results.
- We used accuracy as the main metric for training.
- For the optimizer, we tried gradient descent, stochastic gradient descent and adam. Adam gave better results.

- For the loss function, we used binary cross entropy.

6. RESULTS

	F1-score
Word2Vec with MLP	0.778
Word2Vec with LSTM	0.794
Word2Vec with SVM	0.684
FastText with MLP	0.746
FastText with LSTM	0.794
FastText with SVM	0.694
GloVe with MLP	0.768
GloVe with LSTM	0.782
GloVe with SVM	0.81
TF-IDF with MLP	0.758

LSA (TF-IDF + SVD) + Logistic Regression	0.644

	Mean of Cross validation score
Word2Vec with XGBoost	0.6876
FastText with XGBoost	0.6864
GloVe with XGBoost	0.7989

	Coherence value
Word2Vec with LDA (LDA2Vec)	0.3096

	Accuracy
BERTweet with Logistic Regression	0.80656

- Out of all the models tested, BERTweet with Logistic regression gave the best accuracy.
- Out of all the topic classifiers (MLP, LSTM, SVM) used with the word embeddings (Word2Vec, FastText, GloVe), LSTM turned out to be the best classifier.
- XGBoost worked the best with GloVe as compared to Word2Vec and FastText.
- Supervised techniques like MLP, LSTM, XGBoost, SVM gave better results than unsupervised techniques like LDA and LSA

7. CONCLUSION

In order to meet our objective we tried various classifiers and among those LDA gave the least accuracy, as was expected because it is an unsupervised modelling technique and a probabilistic model and hence the words that describe a topic can be present in both the classes, that is, disaster related and non-disaster related. LSA, which is a dimensionality reduction technique on combining with the classifier, logistic regression, also gave results similar to LDA. XGBoost and SVM were not able to capture the semantic details behind the tweets, so even they gave less accuracy. After these, we experimented with two deep learning models, MLP and LSTM and their performances were almost the same and better than the previously used techniques. LSTM performed slightly better than MLP as it can detect the long term semantic relationship in tweets. Therefore, supervised techniques like LSTM, MLP performed better than unsupervised techniques like LDA and LSA. Finally we worked with BERTweet. Out of all the models using the transformer based model, BERTweet gave the highest F-score.

8. REFERENCES

1. Singh, J.P., Dwivedi, Y.K., Rana, N.P. et al. Event classification and location prediction from tweets during disasters. Ann Oper Res 283, 737–757 (2019). https://doi.org/10.1007/s10479-017-2522-3 https://link.springer.com/article/10.1007/s10479-017-2522-3
2. Identifying and Categorizing Disaster-Related Tweets by Kevin Stowe, Michael Paul, Martha Palmer, Leysia Palen, Ken Anderson at Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media https://aclanthology.org/W16-6201.pdf
3. Real or Not? NLP with Disaster Tweets by Chandan Mannem, Mahalavanya Sriram, Aparajitha Sriram
4. Bagheri H , Islam Md J (2017). Sentiment analysis of twitter data. Computer Science Department Iowa State University,United States of America.
5. Goswami, Shriya and Raychaudhuri, Debaditya, Identification of Disaster-Related Tweets Using Natural Language Processing: International Conference on Recent Trends in Artificial Intelligence, IOT, Smart Cities & Applications (ICAISC-2020) (May 26, 2020). Available at SSRN: https://ssrn.com/abstract=3610676 or http://dx.doi.org/10.2139/ssrn.3610676
6. Tweets Classification with BERT in the Field of Disaster Management Guoqin Ma https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/custom/15785631.pdf
7. Disaster Analysis Through Tweets by Himanshu Shekhar, Shankar Setty https://www.researchgate.net/publication/281768488_Disaster_Analysis_Through_Tweets
8. S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, “Microblogging during two natural hazards events: what twitter may contribute to situational awareness,” in Proceedings of the SIGCHI conference on human factors in computing systems. ACM, 2010, pp. 1079–1088.
9.] Hasan A , Moin S, Karim A , Shamshirband S (2018). Machine learning based sentiment analysis for twitter accounts. Mathematical and Computational Applications, 23(11), 1-15.
10. Ikonomakis Emmanouil K , Kotsiantis S , Tampakas V (2005). Text classification using machine learning techniques .Wseas Transactions on Computers, 4(8): 966-974.