BANA 7047 – Data Mining II(001)

MINING TWEETS FOR DISASTER MANAGEMENT

SPRING 2020

Submitted by Group 9:

Priyanka Pavithran

Varun Varma

Vipul Mayank

University of CINCINNATI

ABSTRACT

Twitter is a free social networking service that allows people to broadcast short posts called tweets. Tweets can serve as a data source to detect real time disaster events that can be utilized to dig significant information for crisis reaction and relief operation. It is easy for a human to identify whether a tweet is related to disaster or not as some words can be used both in disaster as well as non disaster tweets but very difficult for a machine to identify.

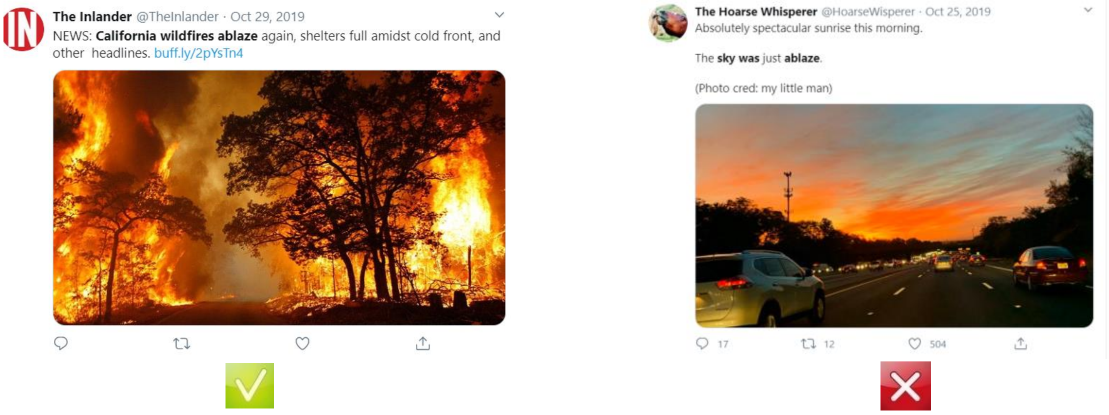
INTRODUCTION

Twitter has become an important communication channel in times of emergency. The ubiquitousness of smartphones enables people to announce an emergency they're observing in real-time. Because of this, more agencies are interested in programatically monitoring Twitter (i.e. disaster relief organizations and news agencies).

But, it's not always clear whether a person's words are actually announcing a disaster. Take this example:

TWEET A: "The sky was just **ABLAZE**"

TWEET B: "California Wildfires **ABLAZE** again, shelters full amidst cold front, and other headlines."



The Inlander @TheInlander · Oct 29, 2019
NEWS: **California wildfires ablaze** again, shelters full amidst cold front, and other headlines. buff.ly/2pYsTn4

The Hoarse Whisperer @HoarseWisperer · Oct 25, 2019
Absolutely spectacular sunrise this morning.

The **sky was** just **ablaze**.

(Photo cred: my little man)

Figure 1: Checking two tweets with same keyword

The author explicitly uses the word "ABLAZE" but means it metaphorically. This is clear to a human right away, especially with the visual aid. But it's less clear to a machine.

Hence, we have build a machine learning model (in Python) that predicts which Tweets are about real disasters and which one's aren't. We have used GloVe technique to convert the words into its vector form and applied different supervised learning models on them to predict the result. Out of all the models created using various machine learning algorithm, Boosting (XGBoost) gives us the best result in all parameters.

DATA SOURCE

The dataset is obtained from an ongoing Kaggle Competition- Real or Not? NLP with Disaster Tweets. Link can be found here: https://www.kaggle.com/c/nlp-getting-started

EXPLORATORY DATA ANALYSIS

We start with counting the frequency of disastrous tweets and non-disastrous tweets.

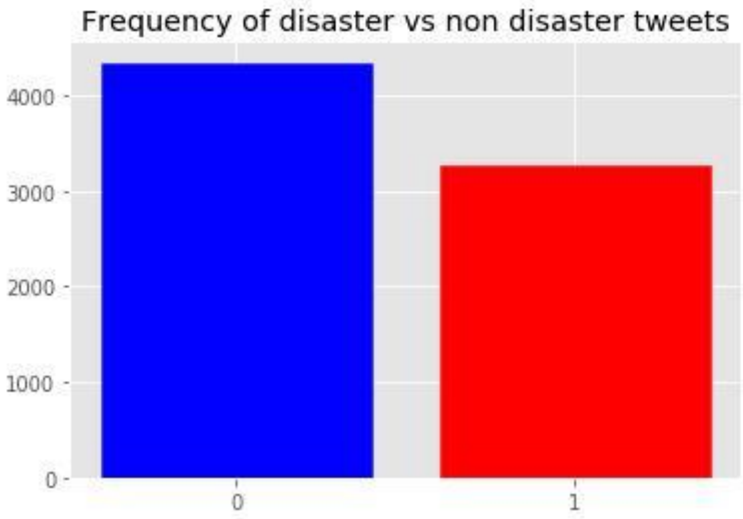


Figure 2: Frequency of Disaster vs Non Disaster Tweets

From the above graph, we see that there are 4342 tweets which are non disasterous while 3271 tweets are classified as disasterous. We will use the same color throughout the analyis. (Red=1=Disaster Tweet, Blue=0= Non Disaster Tweet)

Next, we see the trend of the characters in tweets which are classified as disaster as well as non disaster.

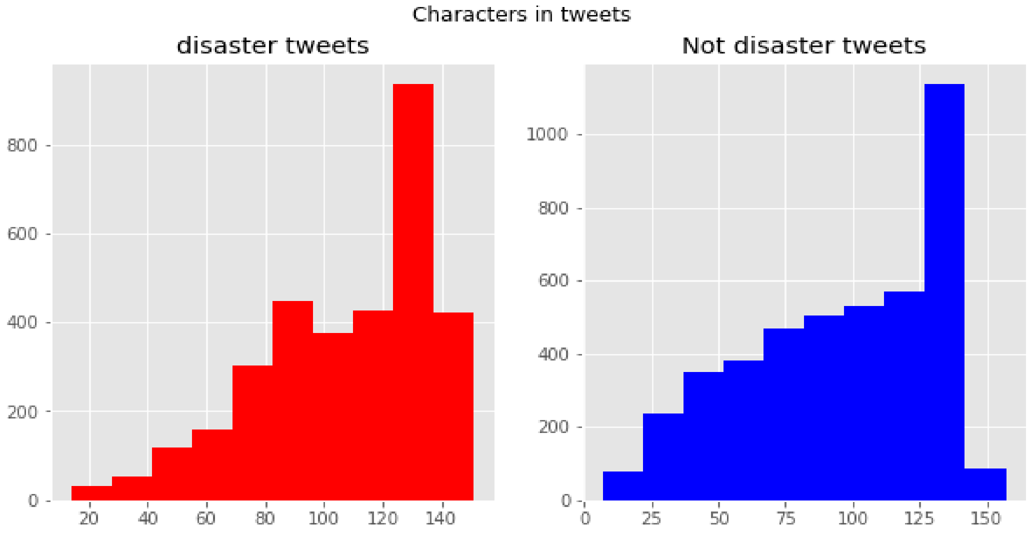


Figure 3: Trends of Characters in Tweets

From the above graph, we see that in both cases of disaster vs non disaster tweets, most of the tweets have characters around 130.

We also see how the trend of words per tweet varies in tweets which are classified as disaster as well as non disaster.

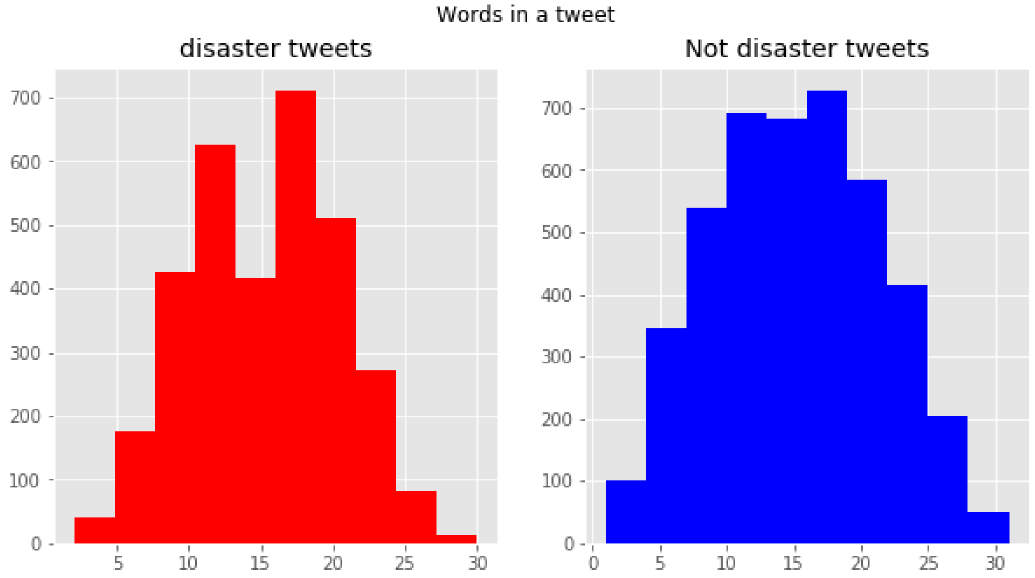


Figure 4: Trends of Words in a tweet

From the above graph, we see that in case of disaster tweets, most of the tweets have words around 17 where as in non disaster tweets , most number of tweets have words ranging from 10 to 18.

Now, let’s look into stopwords. Stopwords can be described as words in a sentence that do not provide any meaningful value. Hence we should remove them from further analysis. Before that, let us check the top 10 stop words in both disaster as well as non disaster tweets.

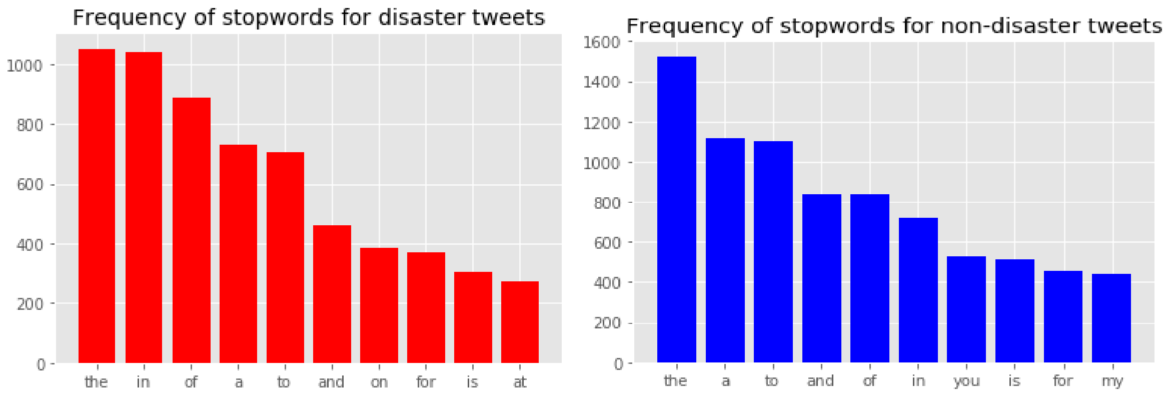


Figure 5: Frequency of Stopwords in Tweets

From the above graph, we see that the stop words are quite similar in both the group with ‘the‘ as the most repeated word.

Similar to stop words, let us look into the special characters. In general they don’t provide any meaning during analysis.

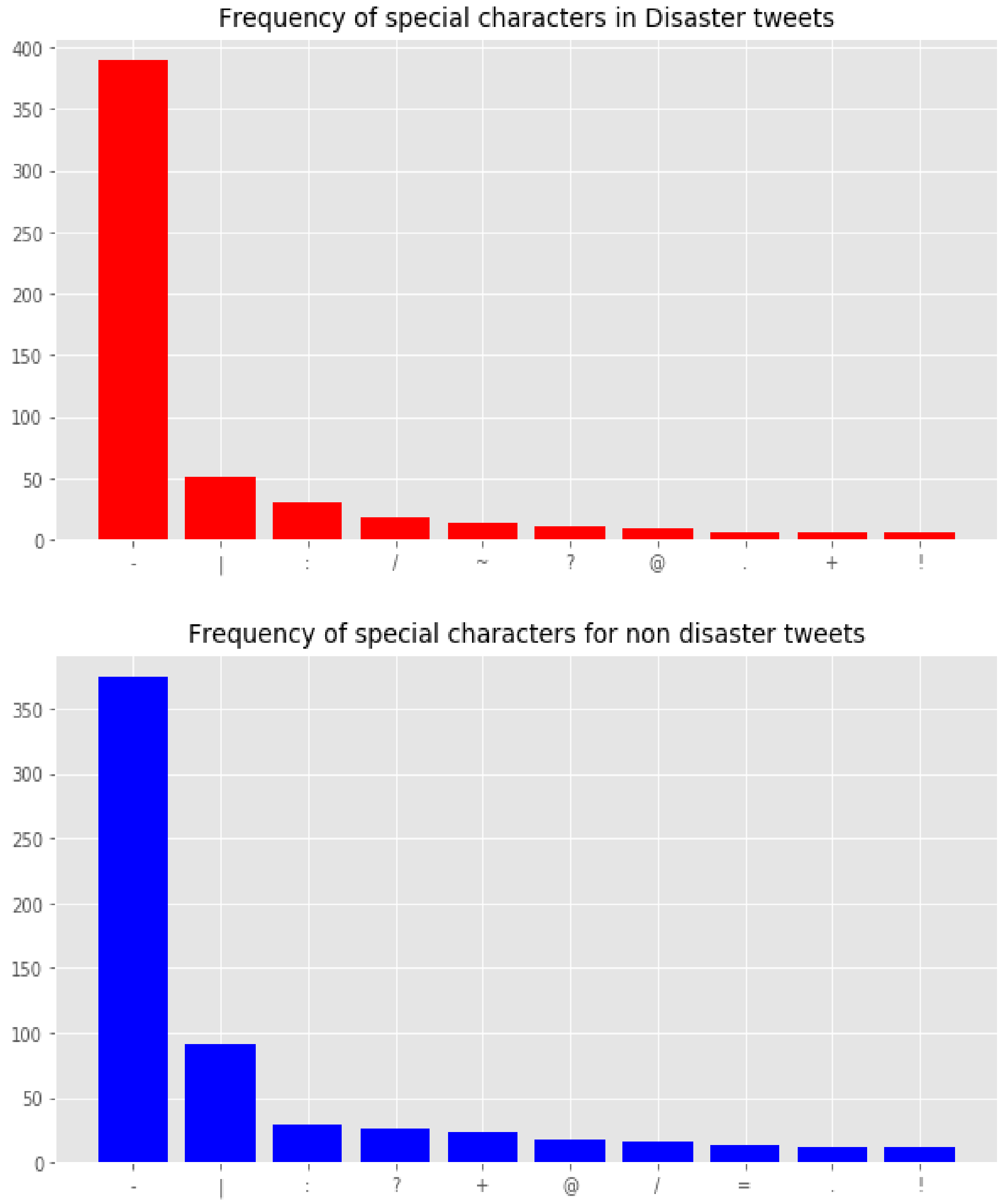
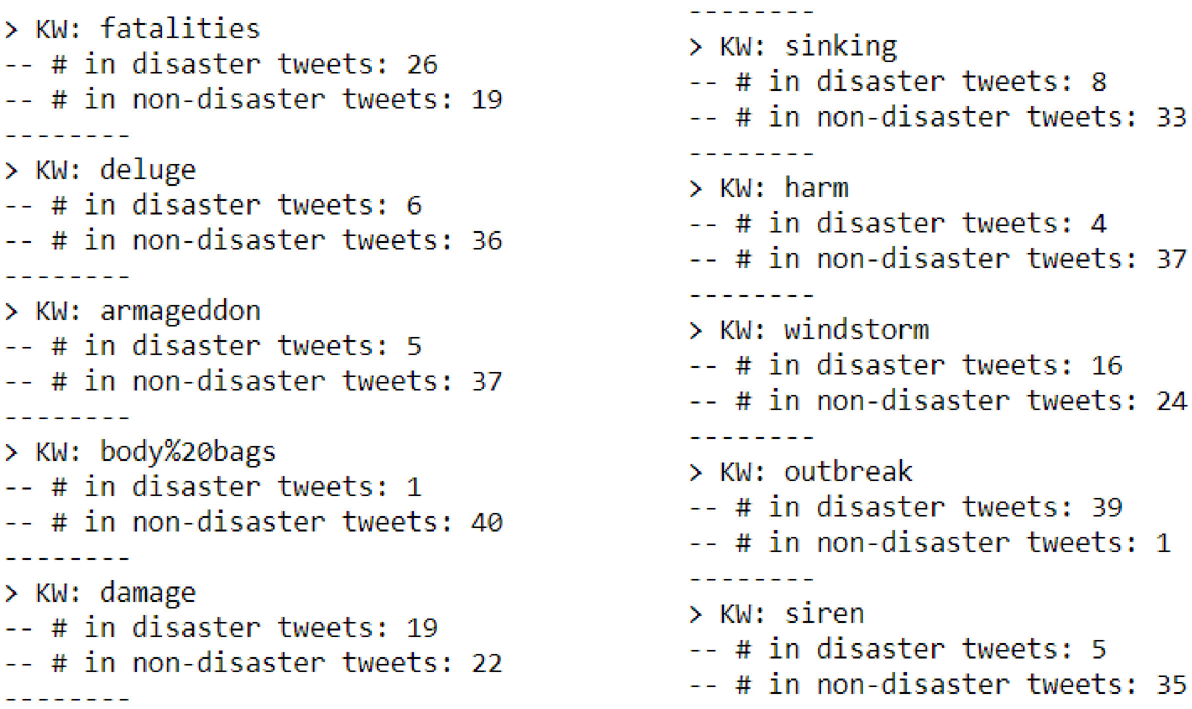


Figure 6: Frequency of Special Characters in Tweets

From the above graphs, we see that special characters are more used in disaster tweets compared to non disaster tweets. Also it is important to note the special characters remain same in both group hence don’t provide much information.

Let us look into the top 10 keywords which are appearing in both disaster tweets as well as non disaster tweets.



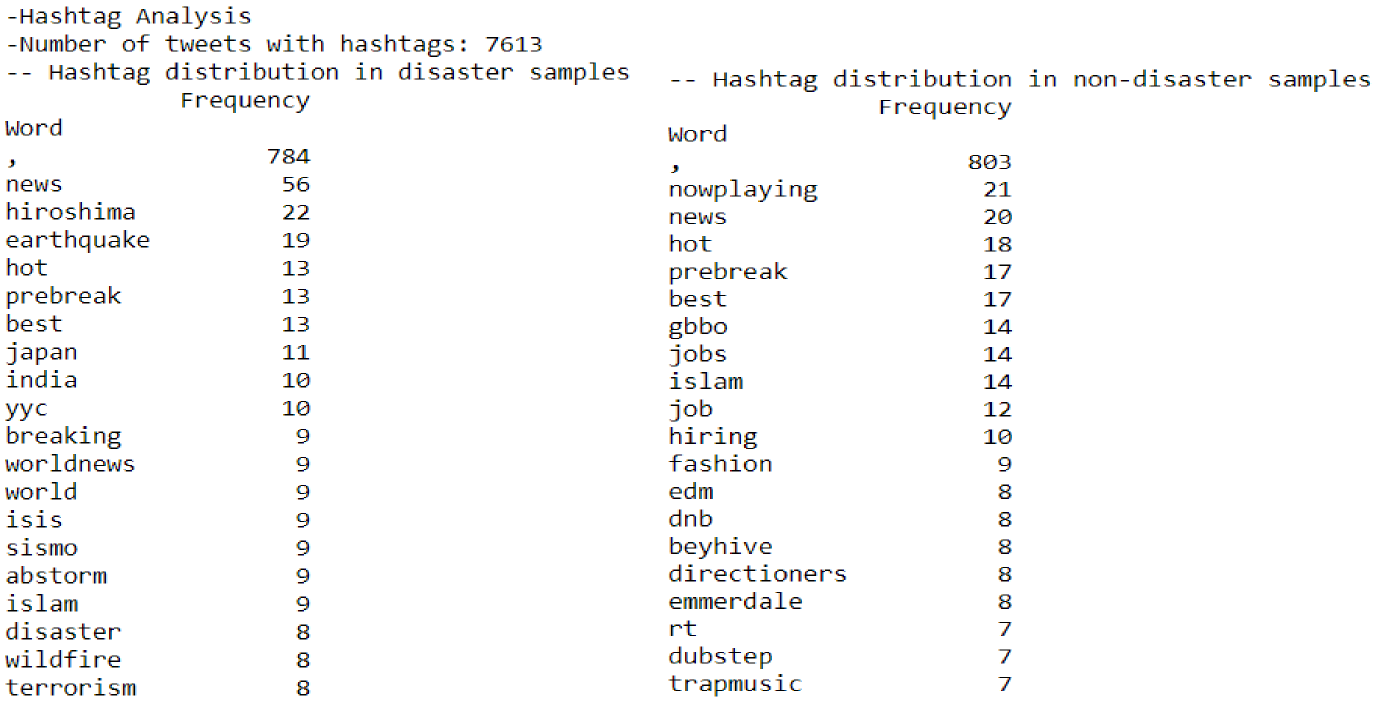
```
> KW: fatalities
-- # in disaster tweets: 26
-- # in non-disaster tweets: 19
--------
> KW: deluge
-- # in disaster tweets: 6
-- # in non-disaster tweets: 36
--------
> KW: armageddon
-- # in disaster tweets: 5
-- # in non-disaster tweets: 37
--------
> KW: body%20bags
-- # in disaster tweets: 1
-- # in non-disaster tweets: 40
--------
> KW: damage
-- # in disaster tweets: 19
-- # in non-disaster tweets: 22
--------
```

```
--------
> KW: sinking
-- # in disaster tweets: 8
-- # in non-disaster tweets: 33
--------
> KW: harm
-- # in disaster tweets: 4
-- # in non-disaster tweets: 37
--------
> KW: windstorm
-- # in disaster tweets: 16
-- # in non-disaster tweets: 24
--------
> KW: outbreak
-- # in disaster tweets: 39
-- # in non-disaster tweets: 1
--------
> KW: siren
-- # in disaster tweets: 5
-- # in non-disaster tweets: 35
```

Figure 7: Top Keywords

This analysis just affirm the basic idea that by just looking at the word present in a tweet, the tweet cannot be classified as disaster or non disaster.

Hashtags plays an important role in a tweet as it supports the idea discussed in the tweet. Small analysis done on the hashtags, to check it's possible discriminator capability for this task.



```
-Hashtag Analysis
-Number of tweets with hashtags: 7613
-- Hashtag distribution in disaster samples
            Frequency
Word
,                  784
news                56
hiroshima           22
earthquake          19
hot                 13
prebreak            13
best                13
japan               11
india               10
yyc                 10
breaking             9
worldnews            9
world                9
isis                 9
sismo                9
abstorm              9
islam                9
disaster             8
wildfire             8
terrorism            8
```

```
-- Hashtag distribution in non-disaster samples
              Frequency
Word
,                  803
nowplaying          21
news                20
hot                 18
prebreak            17
best                17
gbbo                14
jobs                14
islam               14
job                 12
hiring              10
fashion              9
edm                  8
dnb                  8
beyhive              8
directioners         8
emmerdale            8
rt                   7
dubstep              7
trapmusic            7
```

Figure 8: Frequency of Hastags in Tweets

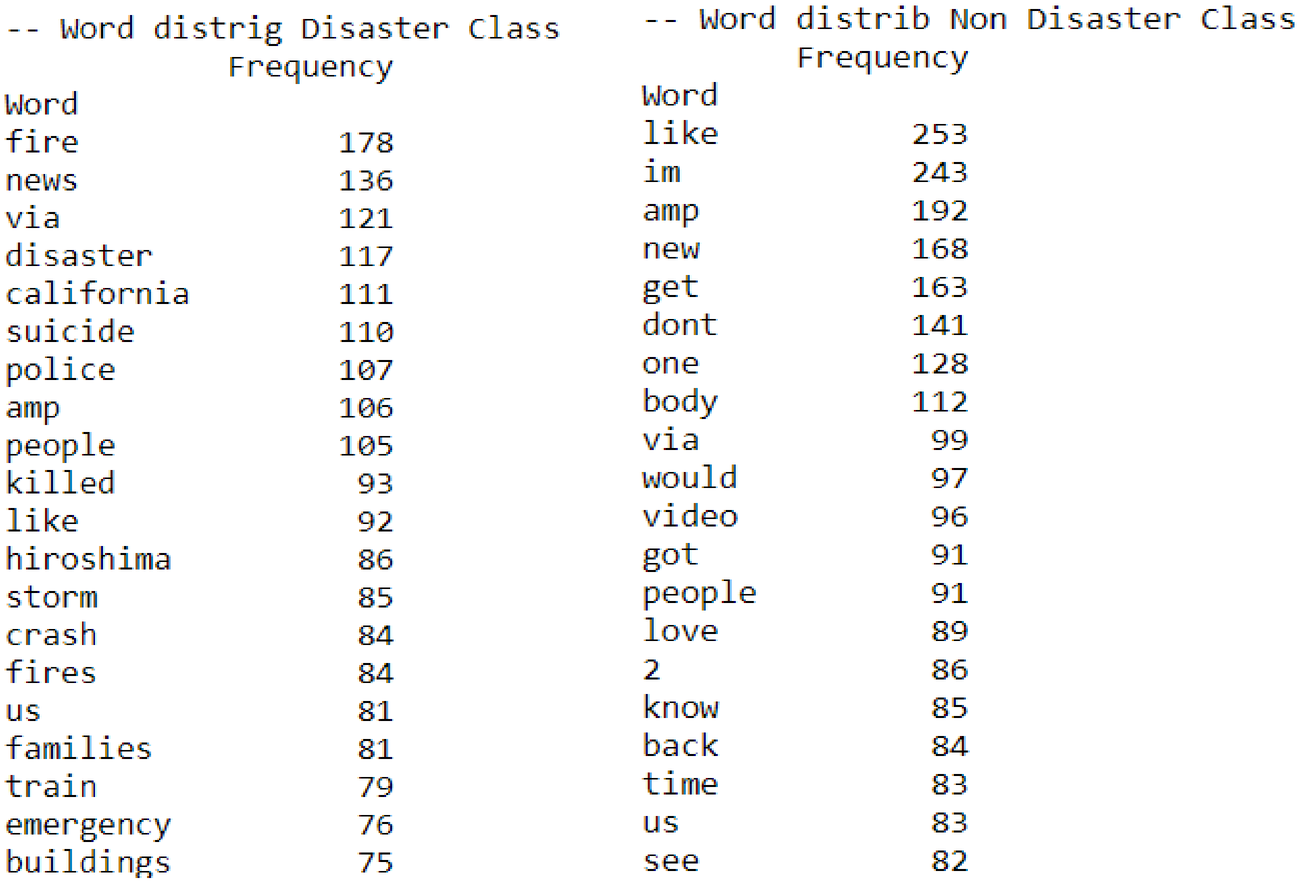
There is too much intersection between hashtag in positive and negative samples, meaning that an #hashtag approach will not work that well in classification.

DATA CLEANING

We clean the data of the following parts as it doesn't add much value to the classification:

- stopwords
- URL
- HTML
- emoticons
- punctuation

After cleaning the data, now we check the top words based on frequency in both disaster as well as non disaster tweets.



```
-- Word distrig Disaster Class
            Frequency
Word
fire              178
news              136
via               121
disaster          117
california        111
suicide           110
police            107
amp               106
people            105
killed             93
like               92
hiroshima          86
storm              85
crash              84
fires              84
us                 81
families           81
train              79
emergency          76
buildings          75
```

```
-- Word distrib Non Disaster Class
        Frequency
Word
like          253
im            243
amp           192
new           168
get           163
dont          141
one           128
body          112
via            99
would          97
video          96
got            91
people         91
love           89
2              86
know           85
back           84
time           83
us             83
see            82
```

Figure 9: Frequency of Top Words after Data Cleaning

From the above plot, we see that the top words of disaster tweets are more negative words when compared to words of non disaster tweets.

Similarly, we check the bigrams and trigrams of the cleaned data.

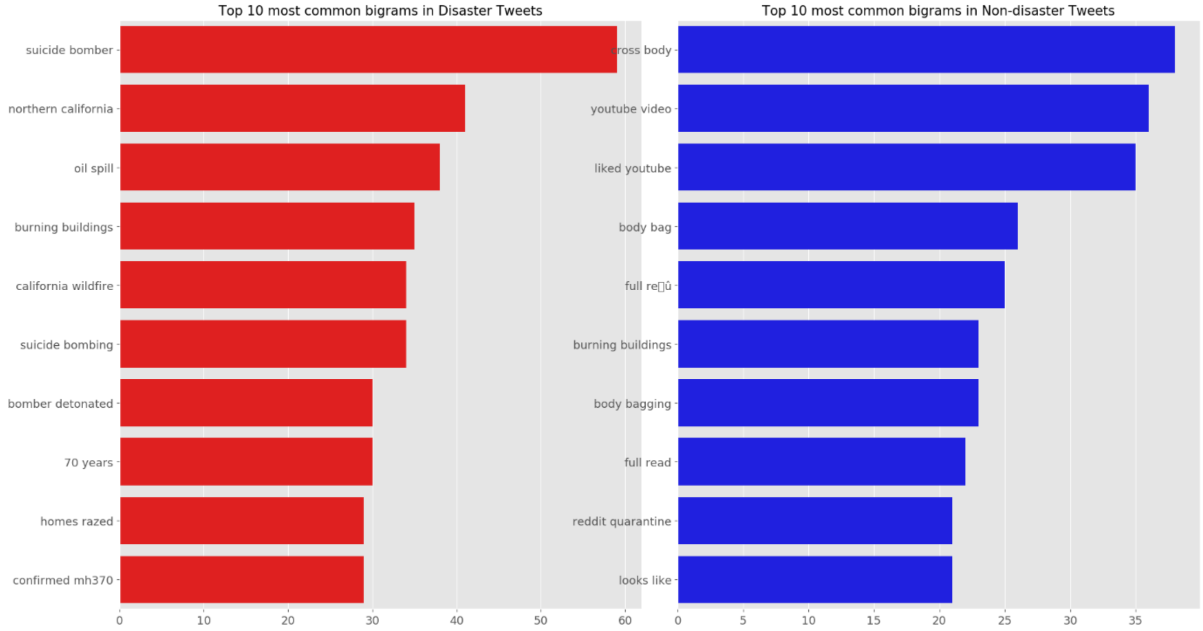


Figure 10: Frequency of Bi-grams words

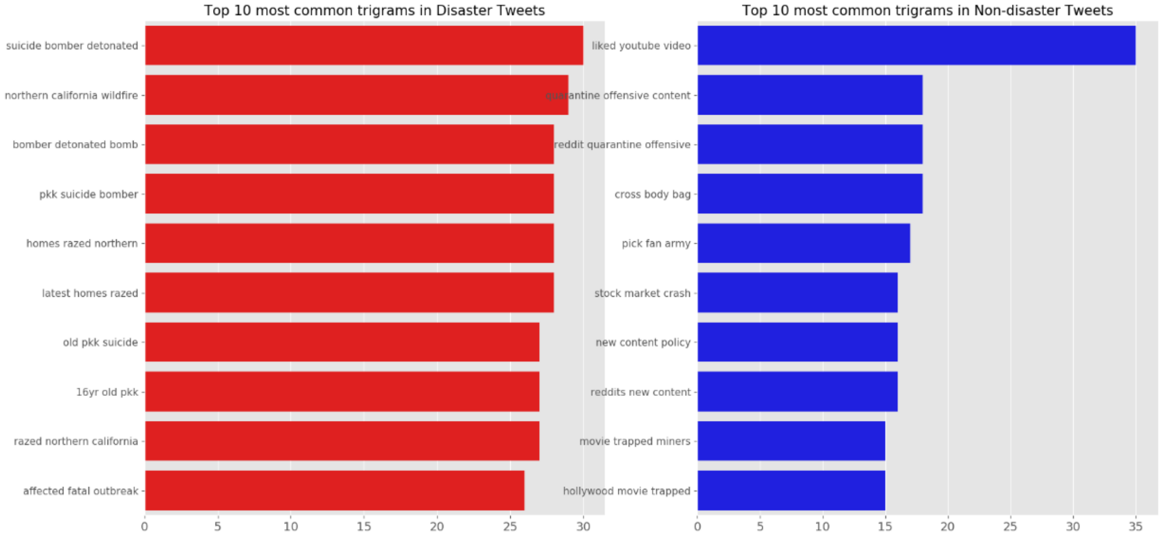


Figure 11: Frequency of Tri-grams words

DATA PREPARATION

We are now ready to do modeling. For that, we will divide the dataset into training and testing(80% and 20% respectively).

We used Global Vector(GloVe) technique of word embedding method to convert each word into a 300 dimensional vector of number.

GloVe is a word vector technique that put words to a nice vector space, where similar words cluster together and different words repel.

The way GloVe predicts surrounding words is by maximizing the probability of a context word occurring given a center word by performing a dynamic logistic regression.

The advantage of GloVe is that, unlike Word2vec, GloVe does not rely just on local statistics (local context information of words) but incorporates global statistics (word co-occurrence) to obtain word vectors.

K-NEAREST NEIGHBOURS(KNN)

We run a loop to vary the number of nearest neighbour in the algorithm and measure the accuracy of training and testing data.

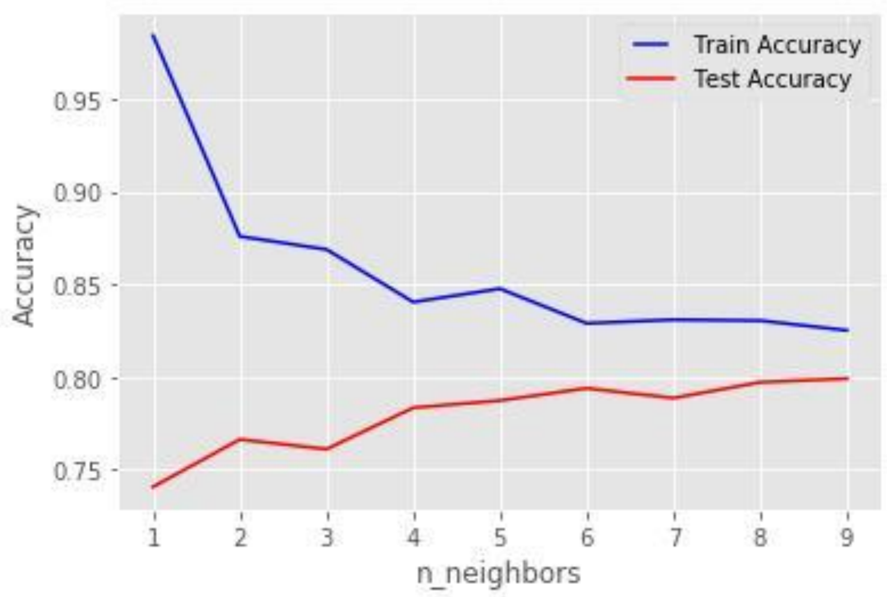


Figure 12: Accuracy vs Nearest Neighbor Graph

From the above graph, we fix the tuning parameter of k as 9 and train the model to evaluate the accuracy on both training and testing dataset.

We also evaluate CV score on the entire dataset. The results are as follow:

Parameters	Value
In Sample Accuracy	0.825
Out of Sample Accuracy	0.799
F1 Score on Test Data	0.758
Cross Validation F1 Score(k=5)	0.744

Table 1: Model Parameters of KNN Model

LOGISTIC REGRESSIONS

After multiple attempt, we fix the asymmetrical cost as 4:1 and hence pcut as 0.25. One of the main task is to reduce the false positive values as we rather classify non-disaster tweets as disaster tweets than classifying the disaster tweets as non disaster tweets.

We calculate the model accuracy on in sample data and out of sample data and cross validation score on full data.

Parameters	Value
In Sample Accuracy	0.803
Out of Sample Accuracy	0.757
F1 Score on Test Data	0.839
Cross Validation Score(k=5)	0.666

Table 2: Model Parameters of Logistic Regression

CLASSIFICATION DECISION TREE

In order to prune the tree, we run a loop from 0 to 30 depth with two methods gini and entropy and compare the accuracy.

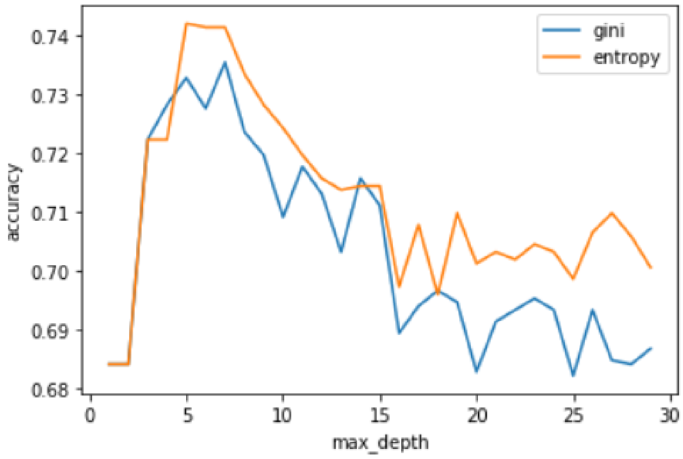


Figure 13: Accuracy vs No. of Nodes for Gini and Entropy Method

From the above graph, we can see that the accuracy is maximum for max_depth =5 and entropy method. We use these tuning parameters to train the model and find the accuracy on the training and testing data along with cross validation score on the full data.

Parameters	Value
In Sample Accuracy	0.776
Out of Sample Accuracy	0.741
F1 Score on Test Data	0.663
Cross Validation Score(k=5)	0.720

Table 3: Model Parameters of Classification Tree Model

BOOSTING

We use XGBoost technique to make a model by varying the learning rate and n_estimator.

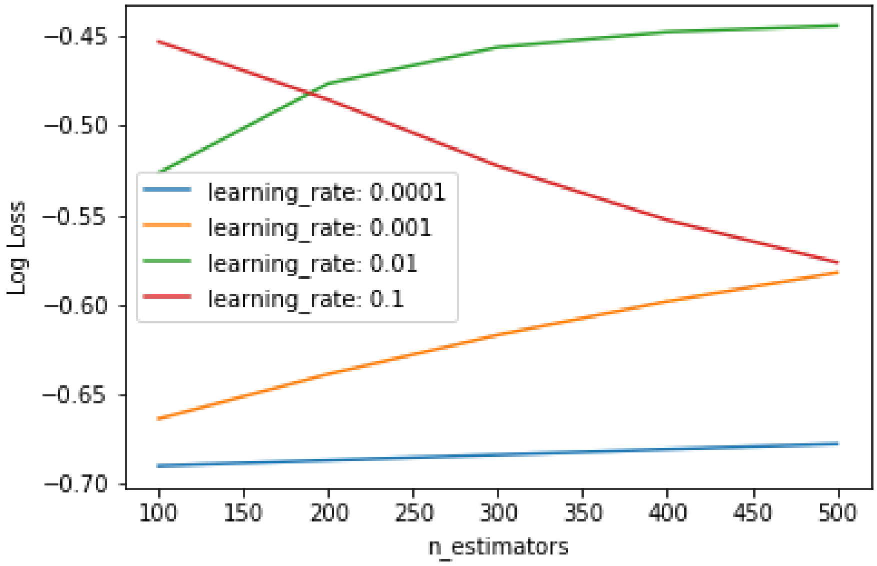


Figure 12: Log Loss vs No. of Trees for Different Learning Rate

From the above graph, we fix the tuning parameters by choosing learning rate as 0.01 and number of trees as 500 because the log loss is minimum these values.

These parameters are used to train the model and find the accuracy on the training and testing data along with cross validation score on the full data.

Parameters	Value
In Sample Accuracy	0.952
Out of Sample Accuracy	0.811
F1 Score on Test Data	0.764
Cross Validation F1 Score(k=5)	0.784

Table 4: Model Parameters of Advanced Boosting Tree Model

VOTING CLASSIFIER MODEL

A Voting Classifier is a machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of chosen class as the output.

It simply aggregates the findings of each classifier passed into Voting Classifier and predicts the output class based on the highest majority of voting. The idea is instead of creating separate dedicated models and finding the accuracy for each them, we create a single model which trains by these models and predicts output based on their combined majority of voting for each output class.

Hence we make a voting classifier model based on all the previous 4 models we have created i.e. KNN, Logistic Regression, Classification Tree and Boosting.

The results obtained from the above combined model on train, test and full data are as follow:

Parameters	Value
In Sample Accuracy	0.902
Out of Sample Accuracy	0.805
F1 Score on Test Data	0.663
Cross Validation F1 Score(k=5)	0.709

Table 5: Model Parameters of Voting Classifier Model

CONCLUSION

After initial analysis of removing the stop words, special characters, hashtag symbols, emoticons, html, url and punctuations, the whole dataset was divided into 80%(training) and 20%(testing).

We use GloVe technique to convert the words into 300 dimensional vectors of number. After that we have created 5 different models. We now evaluate the various models created by comparing the model parameters and the results are as follow:

Parameters	KNN Model	Logistic Regression	Classification Tree	Boosting	Voting Classification Model
In Sample Accuracy	0.825	0.803	0.776	0.952	0.902
Out of Sample Accuracy	0.799	0.757	0.741	0.811	0.805
F1 Score on Test Data	0.758	0.839	0.663	0.764	.663
CV F1 Score(K=5)	0.744	0.666	0.720	0.784	0.709

From the above results, we see that Boosting(XGBoost) technique stands out in all analysis compared to the result of the other models.

Hence we will go with Boosting as our final model.

TEST ON A SAMPLE TWEET

We have taken two tweets with the same keyword ‘ablaze’ and passed in our model to check how it performs.

- Tweet A:
  Birmingham Wholesale Market is ablaze BBC News –
  Fire breaks out at Birmingham's Wholesale Market http://t.co/irWqCEZWEU
  Classified as a **DISASTER**

- Tweet B:
  Not a diss song. People will take 1 thing and run with it.
  Smh it's an eye opener though. He is about 2 set the game ablaze
  @CyhiThePrynce
  Classified as a **NON-DISASTER**

BIBLIOGRAPHY

- https://www.datacamp.com/community/tutorials/text-analytics-beginners-nltk
- https://medium.com/towards-artificial-intelligence/text-mining-in-python-steps-and-examples-78b3f8fd913b
- https://www.coursera.org/learn/python-text-mining
- https://www.analyticsvidhya.com/blog/2018/02/the-different-methods-deal-text-data-predictive-python/
- https://www.kaggle.com/philculliton/nlp-getting-started-tutorial