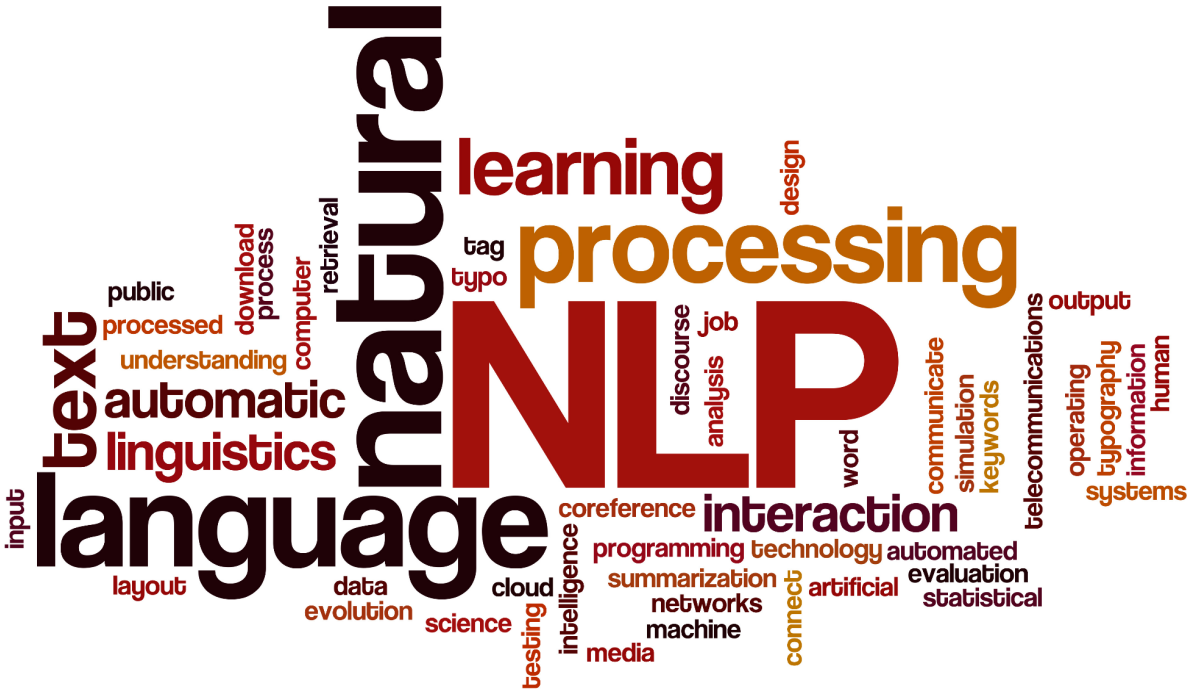


Natural Language Processing

Group Project

EN502 - Machine Learning Course

School of Engineering

Jawaharlal Nehru University

Team Details

NAME	ROLL NO	CONTACT	EMAIL
Santosh Jonnakuti	19/11/EC/043	8328164411	santoshjonnakuti@gmail.com
Chokkari Dinesh	19/11/EC/062	9494204221	dineshdinnu8118@gmail.com
LS Yaswanth Kumar	19/11/EC/059	7032356627	24.yashwanth@gmail.com
Danam Yashwanth	19/11/EC/040	8688375610	yk1539614@gmail.com
Bathini Hemanth	19/11/EC/039	8688270785	hemanthbathini.123789@gmail.com

Team Leader Details

Santosh Jonnakuti

Project Title

Know What's Real

Project Description

Natural Language Processing or NLP is an AI concerned with the interaction between human language and computers. NLP is all about analyzing and representing human language computationally. It equips computers to respond using context just like a human would and the rise of technologies like text and speech recognition, sentiment analysis, and machine-to-human communications, has inspired several innovations.

Twitter is one of the biggest social media platforms where people from different countries and regions can express their thoughts, insights regarding any instance. The ubiquitousness of smartphones enables people to announce an emergency they're observing in real-time. If the news is real it saves so many lives given the gravity of the disaster. Incase of fake news, it spreads faster than the real news and is more concerning. According to the reports given by several surveys, the percentage of fake news increased exponentially during the pandemic.

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.

So, our project aim is to classify the difference between real and fake tweets.

Motivation

As Twitter has become an important communication channel many people are falling in the traps of false tweets about disasters and believing them blindly.Concerns over the problem are global and the impact is long-lasting.The past few decades have witnessed the critical role of misinformation detection in enhancing public trust and social stability.So,this has motivated us to choose this project

Challenges

1. Conceptual words and phrase

   The same words and phrases can have different meanings according to the context of a sentence and many words - especially in English - have the exact same pronunciation but totally different meanings.

   EX: I **ran** to the store because we **ran** out of gas.

   Can I **run** something past you real quick ?

2. The automated detection of misinformation about disasters is difficult to accomplish as it requires the advanced model to understand how related or unrelated the reported information is when compared to real information.
3. Misspelled or misused words can create problems for text analysis. Autocorrect and grammar correction applications can handle common mistakes, but don't always understand the writer's intention.
4. Missing Values

   Some of the columns may have some null values which may lead to some errors in prediction.

Methodology

Steps to be followed to build this model.

1. The Data given in the Kaggle Dataset has following Columns

   `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` - in train.csv only, this denotes whether a tweet is about a real disaster (`1`) or not (`0`)
2. Missing values of columns like keyword will be addressed with a feasible method.
3. The column text will be pre-processed like removing the stopwords(prepositions and other common English words).
4. The verbs in the word will be converted into base verbs like running will be converted into run so that unique values in the columns will be reduced.
5. The model can be done using various Algorithms.

   Logistic Regression

   Naive Bayes Classifier

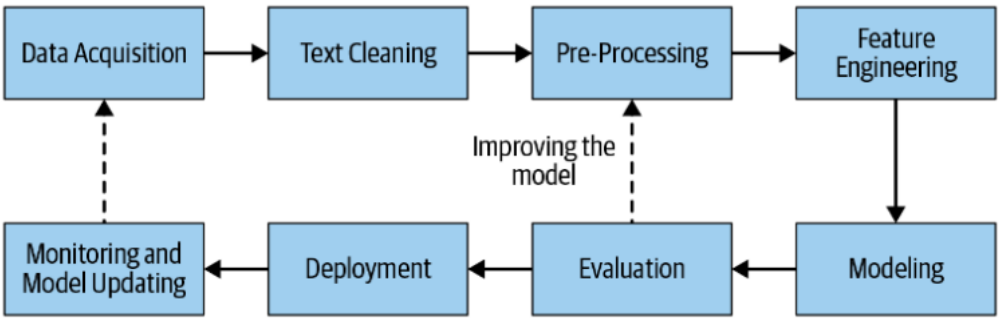
   Support Vector Machine(SVM)

   Random Forest

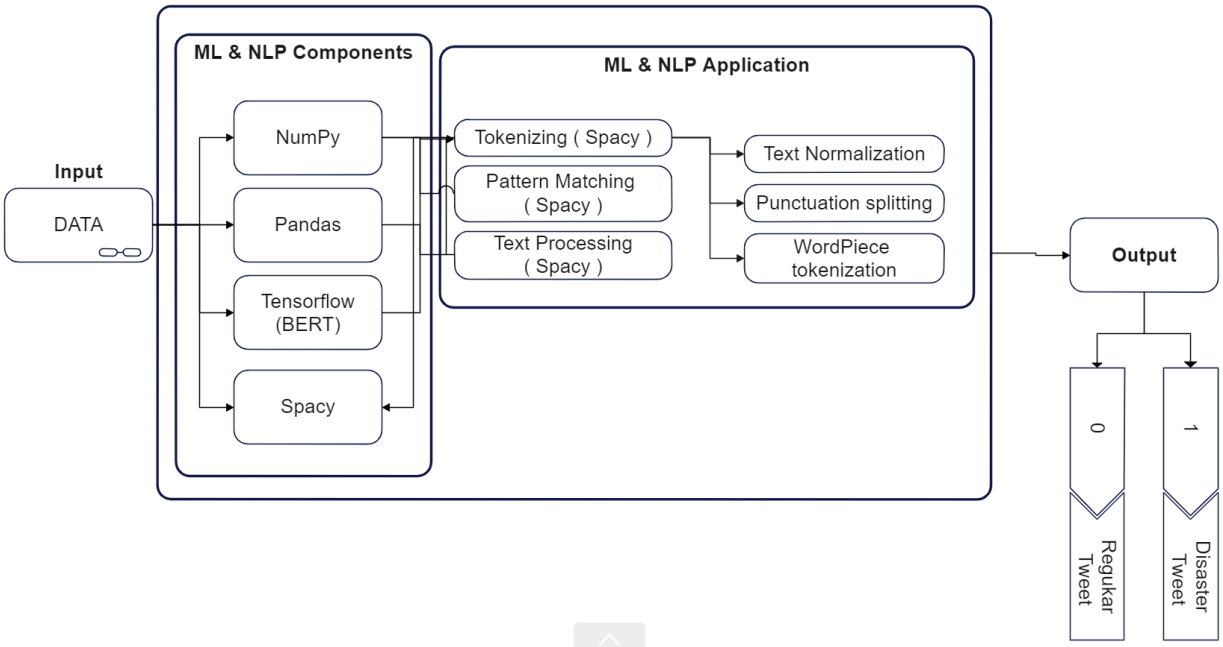
   K Nearest Neighbours (KNN)

6. Using the training data model will be trained using various algorithms to predict the output.
7. After training the model with different algorithms the model with the highest accuracy will be used to deploy the model.

I. Pipeline



II. Architecture Diagram



III. Block Components

This model can be implemented in various methods. Those include

- Logistic Regression
- Naive Bayes Classifier
- Support Vector Machine(SVM)
- Random Forest
- K Nearest Neighbours (KNN)

Results

Expected Results

This model will predict which Tweets are about real disasters and which one's aren't.

For each id in the test set , you must predict 1 if it's a real disaster and 0 otherwise.

ID,TARGET

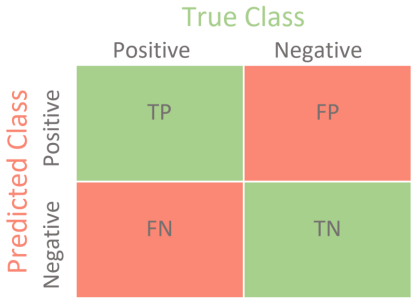
1,1

2,0

Metrics to be used

Confusion Matrix

The confusion matrix is a useful tool used for classification tasks in machine learning with the primary objective of visualizing the performance of a machine learning model.



Precision

In instances where we are concerned with how exact the model's predictions are we would use Precision. The precision metric would inform us of the number of labels that are actually labeled as positive in correspondence to the instances that the classifier labeled as positive.

Precision = TP/ (TP + FP)

TP → TruePositive

FP → FalsePositive

Recall

Recall measures how well the model can recall the positive class (i.e. the number of positive labels that the model identified as positive).

Recall = TP/(TP + FN)

TP → TruePositive

FN → FalseNegative

F1 Score

Precision and Recall are complementary metrics that have an inverse relationship. If both are of interest to us then we'd use the F1 score to combine precision and recall into a single metric.

F1 = 2*precision*recall / (precision + recall)

AUC

The AUC(Area Under the Curve) helps us quantify our model's ability to separate the classes by capturing the count of positive predictions which are correct against the count of positive predictions that are incorrect at different thresholds.

$AUC = (-n_0 * (n_0 + 1)/2)/n_0*n_1$

n_0 → Number of False Information Species

n_1 → Number of True Information Species

Reference

https://www.kaggle.com/c/nlp-getting-started/overview