Stance Detection for Tweets

Kikkuru Sarath Chandra Reddy
Indraprastha Institute of Information Technology
Delhi, India
kikkuru19037@iiitd.ac.in

Kasarla Mani Kumar Reddy
Indraprastha Institute of Information Technology
Delhi, India
kasarla19065@iiitd.ac.in

Koncha Vivek Reddy
Indraprastha Institute of Information Technology
Delhi, India
koncha19038@iiitd.ac.in

Prof.Tanmoy Chakraborty
Indraprastha Institute of Information Technology
Delhi, India
tanmoy@iiitd.ac.in

Abstract—: Extensive research has been carried out on opinion mining by the research community i.e., detection of polarity by classifying the text into one of the three classes: negative, neutral and positive. But very less research has been done towards Stance detection i.e to determine favorability towards given targets of interest. In this project we developed models that will determine automatically whether the author of text is in favour of, or against or None towards a given target.

Problem Statement--: Stance Detection can be explained in many ways. In this task, we define Stance Detection as automatically determining whether the text is in favour or against towards given person of Interest.As an example, for the following pair:

Tweet: A foetus has rights too! Make your voice heard.

Target:Legalization of abortion here stance is negative/against the target.

1. INTRODUCTION

1.1 Overview

Internet has given rise to many social media sites and news sites where anyone can express his/her view regarding any person. These views can be related to product feedback or current issues or events etc.They provide very useful opportunity to use this data to develop automatic systems that detects the opinion towards the target of interest. The main challenge is manual analysis of such large data.Efforts for automatic text analysis and understanding are part of Natural language processing. Most of the NLP methods are statistical based which later makes use of Machine Learning and Information Retrieval techniques.Stance detection is very much related to, but kind of different from sentiment analysis.When dealing with Sentiment analysis, approaches are formulated as determining a piece of text is positive or negative with no regard of specific person of interest. However, in stance detection we express favorability towards given person of interest.

1.2. DATA SET

Stance annotations we use are explained in detail in Mohammad et al. (2016a).We took the dataset from SemEval 2016, Task 6 which is made publicly available.

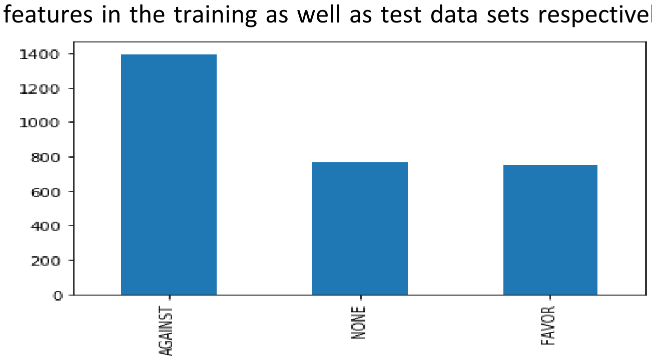
Target of Interests that exist in the dataset are:

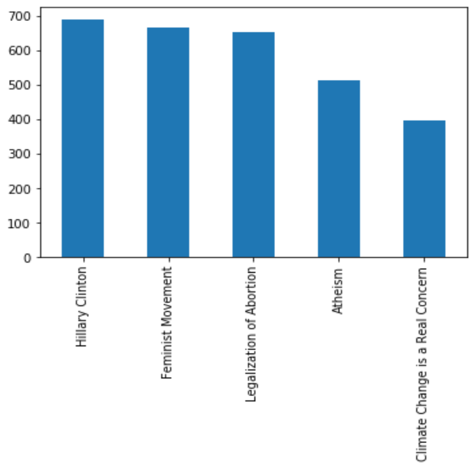
1. Atheism
2. Climate Change is a Real Concern
3. Feminist Movement
4. Hillary Clinton
5. Legalization of Abortion

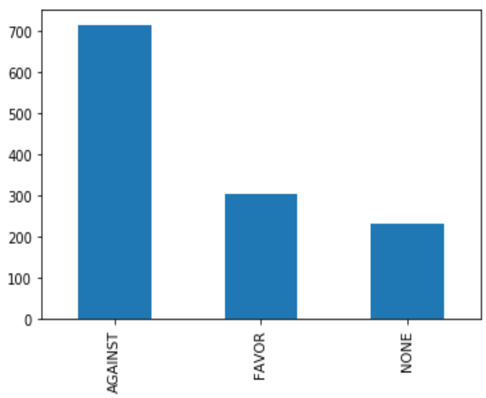
The stance labels are :

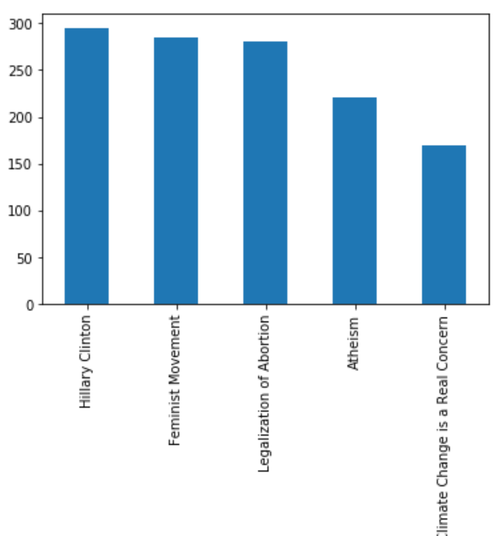
1. Favour: Infer that author is in favour for the target of interest.
2. Against: Infer that author is against the target of interest.
3. None: It includes both neutral and tweets that are not inferred.

Participants were provided with 2,914 labelled tweet-stance pair for five targets. The test data includes 1,249 tweets.following bar plots indicates count of different features in the training as well as test data sets respectively.









1.3. EVALUATION-METRICS

We used the macro-average of the F1 score as the final evaluation metric.
F1-Score = Harmonic mean of precision and recall.

1.4 DATA PREPROCESSING

We have used the following pre-processing techniques on our dataset:

1. Removal of stop-words.
2. Lemmatization.
3. Removal of special Characters.
4. Conversion into lower case.
5. Removal of Alphanumeric Words.
6. Removal of short length words (length <3)
7. Removal of mentions in the tweets

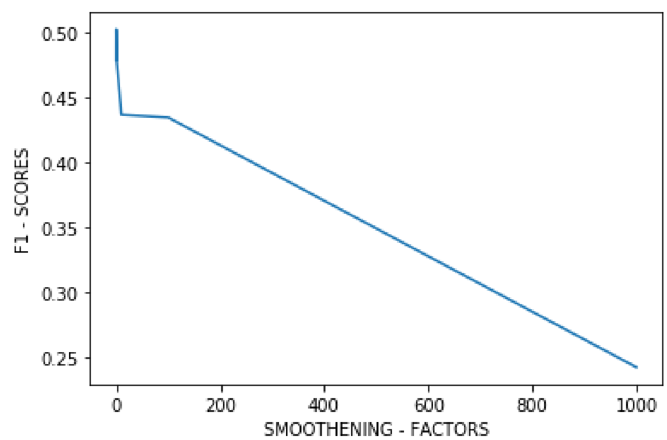
2. METHODOLOGY

2.1 Naive Bayes(baseline)

We have applied Naive Bayes as our baseline model. We used word embeddings such as BAG OF WORDS (BOW) , TFIDF VECTORIZER for encoding the tweet into numerical vectors and used the one-hot encoding technique to encode the corresponding targets.We have mapped the class labels to numeric values as

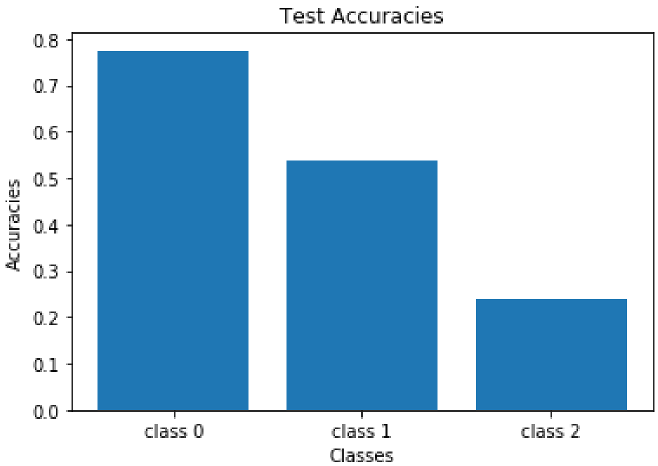
AGAINST - 0
NONE - 1
FAVOR - 2

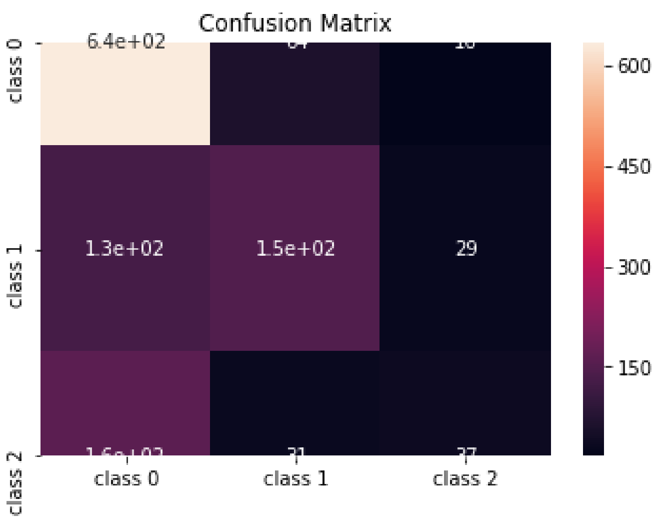
We have considered the smoothing factor (alpha) as our hyper parameter and obtained the best value of f1-score at alpha =0.1. In naive bayes model assumes the conditional independence of features i.e every feature of the data is independent of occurence of other features.The Naive Bayes algorithm will be at its best when the test data is similar to that of training data.
The plot of Smoothening Factor v/s F1-Score is as follows,

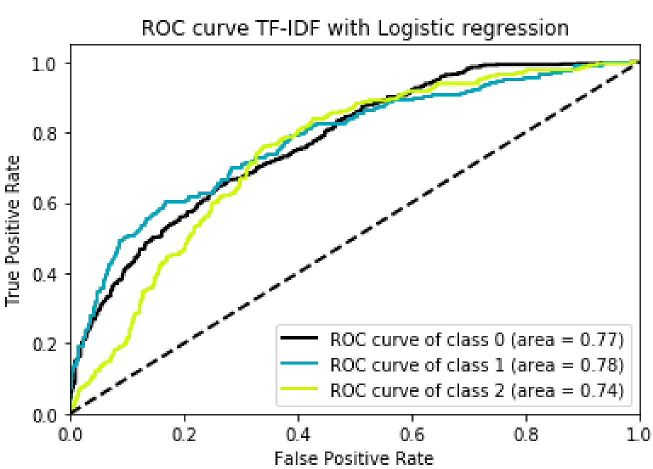


2.2 Logistic Regression:

We use logistic regression when the dependent variable is categorical.The output of model is the estimated probability of text belonging to that particular class. It is a linear model where it tries to find a line that separates the given classes.Since, we have three categorical values we will be using multinomial Logistic Regression. To get prediction from the probabilities we set a threshold value, if the probability is greater than threshold we will assign tweet to that stance.To get the best hyper parameters we used hyperparameter tuning using gridsearch cv. C value is considered as hyperparameter.the following are the results after applying grid search cv.

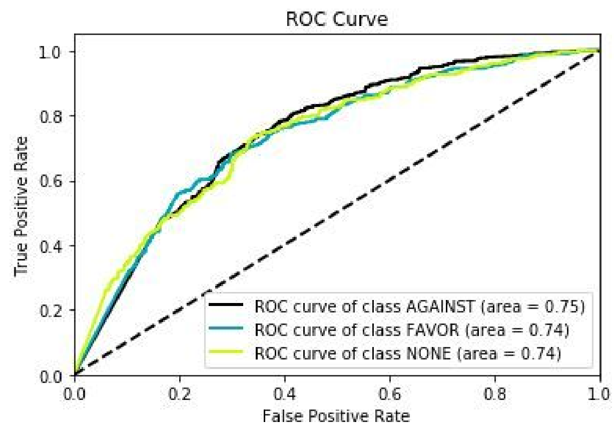


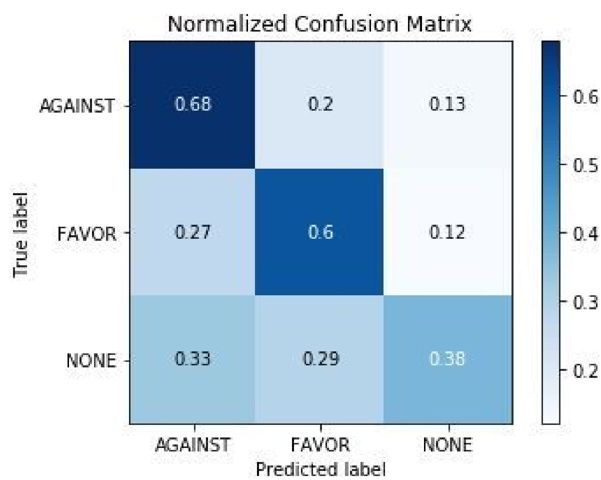




2.3 SVM (Support Vector Machines):

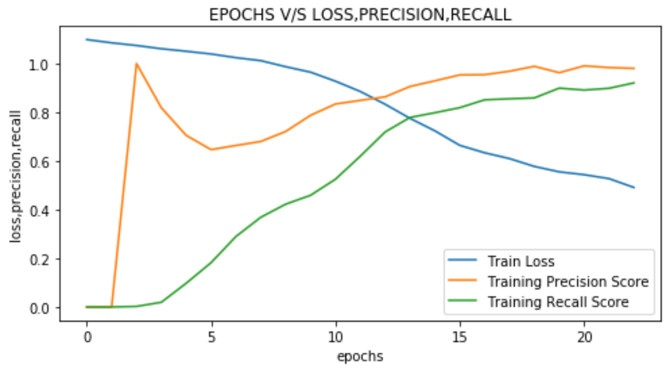
The dataset is preprocessed and vectorized before sending to the SVM model. Since we have three classes which are stance we need to go for multiclass SVM. For faster training time, used Stochastic Gradient Classifier implemented in sklearn. The model was evaluated for best 'alpha'(hyperparameter) across the 'alpha' space with the GridSearchCV tool. Penalty chosen was elastic net which inject regularization of both L1 and L2 with proper proportion among them of the model that leads to better generalization of the model on unseen test tweets





2.4 LSTM (Long Short Term Memory):

In most of the machine learning models the corresponding context of words is not preserved.Whereas, LSTMs preserve these dependencies and are very effective in capturing the contexts between the words of the sentences.LSTM is the special case of recurrent neural network(RNN) which solves the problem of long term dependencies in RNN's. We have built the vocabulary of the training set and ranked the words accordingly to their frequencies and each word in the sentences is replaced by the rank of word in the vocabulary. We have also done padding which makes length of all sentences to be equal.The rank of unseen word in the test set is assigned a particular number or is ignored.The plot of epochs vs (loss,precision,recall) is as follows,



3. RESULTS

F1-SCORES :

- Naive Bayes : 50.2
- Logistic Regression : 54.8
- Support Vector Machines : 56
- LSTM : 74.66

4.CONCLUSION

Approached different ML models and found LSTM to beat the F1-scores of the rest models because the dependencies are being preserved in the LSTM where the later approaches used vectorized operation on tweet for the model to be trained.

5.FUTURE WORK

We want to further analyse other neural network implementations on the same dataset.Sentiment data was released for the tweets we want to explore the field of sentiment classification further.

6.REFRENCES

[1] Semeval-2016 Task 6: Detecting Stance in Tweets. Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. In Proceedings of the International Workshop on Semantic Evaluation (SemEval '16). June 2016. San Diego, California.
[2] Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. 2016. Stance and sentiment in tweets. Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media, In press
[3] Kiritchenko, S., Zhu, X., and Mohammad, S. 2014. Sentiment Analysis of Short Informal Texts. Journal of Artificial Intelligence Research, vol. 50, pages 723-762
[4] Mohammad, Saif M, and Kiritchenko, Svetlana, and Zhu, Xiaodan. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets, In Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013), June 2013, Atlanta, USA.
[5] Murakami, A., and Raymond, R. 2010. Support or Oppose? Classifying Positions in Online Debates from Reply Activities and Opinion Expressions. In Proceedings of the International Conference on Computational Linguistics (ACL), pages 869–875.
[6] Sridhar, Dhanya, Getoor, Lise, and Walker, Marilyn. 2014. Collective Stance Classification of Posts in Online Debate Forums. In Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media, pages 109-117
[7] Thomas, M., Pang, B., and Lee, L. 2006. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 327–335.
[8] Wyner, Adam, and Schneider, Jodi. "Arguing from a Point of View." In Proceedings of the First International Conference on Agreement Technologies. 2012.
[9] Walker, M. A.; Anand, P.; Abbott, R.; and Grant, R. 2012. Stance classification using dialogic properties of persuasion. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 592–596.
[10] Thomas, M., Pang, B., and Lee, L. 2006. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 327–335.
[11]https://machinelearningmastery.com/sequence-classification-lstm-recurrent-neural-networks-python-keras/
[12] https://keras.io/examples/imdb_lstm/
[13]https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html
[14] https://www.aclweb.org/anthology/S16-1063/