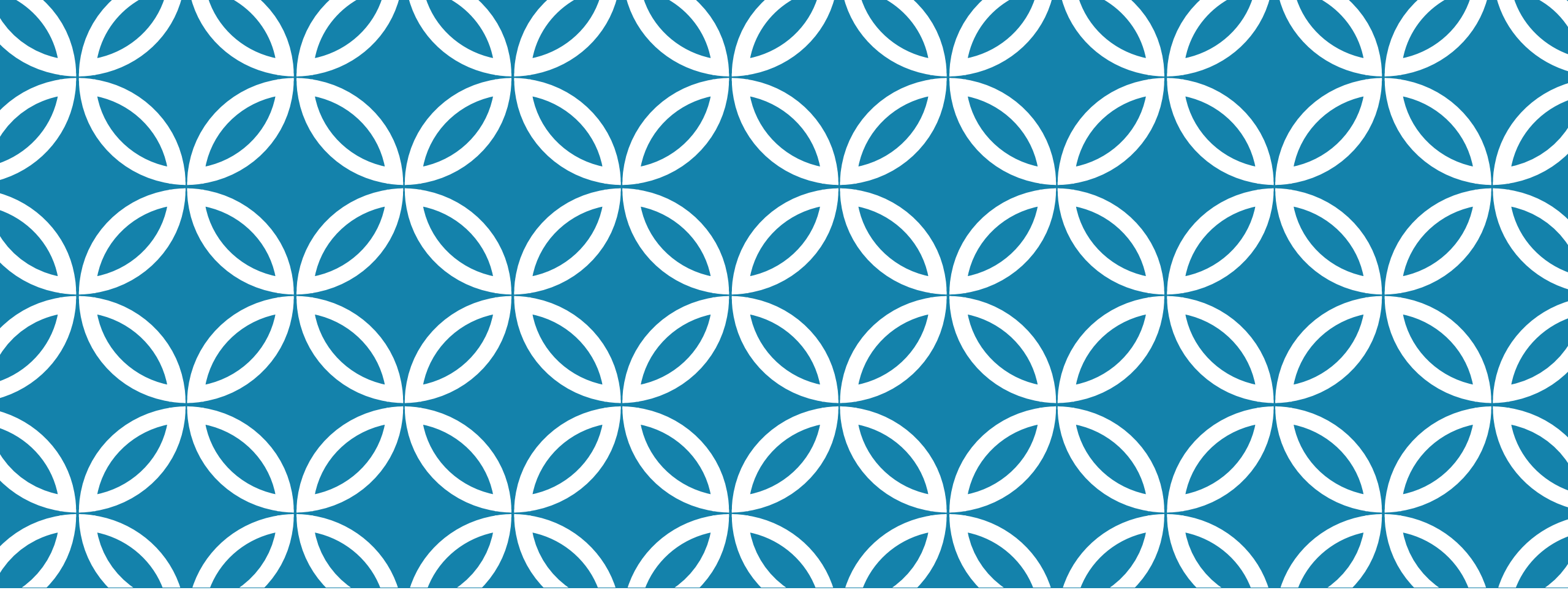


PROJECT TITLE

Lavanya L PES1201700798
Jyosna S PES1201700792

UE17CS333 project submission

ABOUT THE PROJECT

Every day massive amount of data is generated by social media users which is used to analyse their opinion about any event like movie, product or politics. The main aim of this project is to provide a method for analyzing sentiment score in noisy data which includes abbreviated words and slang words i.e Hindi English words in a movie review dataset. This project reports on the design of sentiment analysis, extracting the raw twitter movie reviews from the dataset and pre process it. Results classify user's perception via positive, negative and neutral as sentiment for the review.

UNIQUENESS AND ANALYSIS

- In our project we have used dictionaries to replace the abbreviated words and Hindi-English with suitable english words rather that just removing them so that during the sentiment analysis we do not lose the sentiment conveyed by the users through these words . Most of the research projects do not take care of these pre-processing techniques which affects the accuracy of the whole system .
- Another uniqueness in our project is that we are generating labels for our Twitter dataset rather than manually labelling it. It is one of the unique feature because people are used to process their classifiers on ready-made labelled dataset and its challenging to explore various ways in generating our own labels using lexicon methods and various other approaches.

- We have used two methods to generate labels, first is SentiWordNet and second one is using TextBlob.We tested the labels generated by these two methods by using manually labelled test data and saw the the labels generated by TextBlob is more efficient compared to SentiWordNet.We continued our analysis based on the TextBlob labelling.
- This unique feature which we incorporated has helped us in finding the sentiments precisely for the given tweets and thus feeding it as an input to the classifier.

DATASET SOURCE AND PREPROCESSING DONE

1. Dataset source : The dataset is collected from kaggle which is publicly available and it is a movie review dataset which includes tweets about controversial Bollywood movie after its release.

2. The preprocessing steps performed in our project are:

- **Conversion of Strings to Lowercase** - Normalizing the tweets by converting into lowercase.
- **Expansion of the contracted words:** The contracted words in the tweets are expanded and de-compressed into their respective words.
- **Replacing the Abbreviated/Chat words:** Replaced all the chat word with their respective english words.

- **Removal of stopwords:** Stopwords are the words which dont add any real value to the sentiment of the tweet. Hence all the stopwords are removed.
- **Replacing the Hindi-English words :** There are a lot of informal words in the text which affects the performance of the entire model. Hence we replaced all the hinglish words with their respective English words for better processing.
- **Removal of Punctuations:** The punctuations are removed from the tweets and have been replaced with empty strings which helps in removing the unwanted characters.

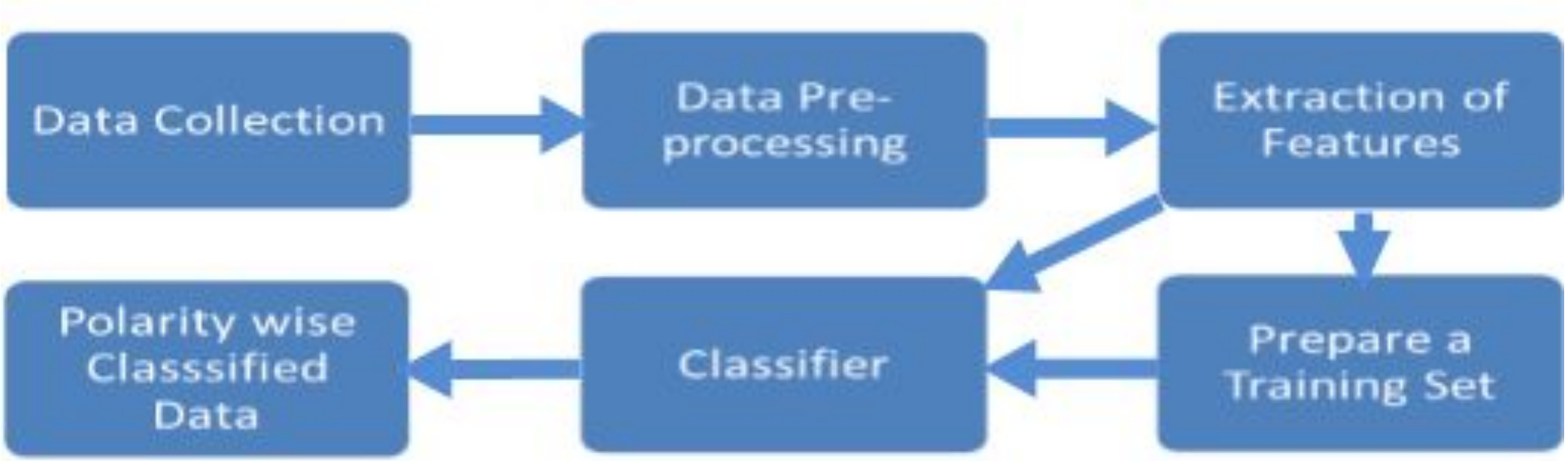
LITERATURE REVIEW

Name of the author	Title of the paper	Year of Publication	Methodology	Performance Parameter	Advantages	Disadvantages
S. Malka, S. Jan and I. A. Shah	Sentiment Analysis based on Abbreviations In Text	2019	Machine Learning Approach, Lexicon Based Approach.	Precision : 89% Recall : 90% F-measure : 91%	-It enhances the accuracy of analysis by dealing with acronyms.	The proposed system does not detect the emotion of words in phrase.
Tajinder Singh and Madhu Kumari	Role of Text Pre-Processing in Twitter Sentiment Analysis	2016	N-grams to find the binding and conditional random fields to check significance of slang word.	Accuracy - 91% Using SVM	Replacing the slang words based on the previous words using n-gram model.	The system accuracy decreases as the size of the dataset increases.
Jagmeet Singh and Dr. Shashi Bhushan	Analysis on Hinglish Opinion Using Multinomial Naive Bayes Algorithm	2016	Used porter stemmer for the Hindi-English word forms and distinction	Precision - 91.4% Recall - 91.3 % True Positive Rate- 91.4%	Enhanced the accuracy of analysis by dealing with Hindi-English words	The system couldn't be executed on large corpus and didn't undergo review of safety experts to judge positivity or negativity of the product.

Literature Review

Name of the author	Title of the paper	Year of Publication	Methodology	Performance Parameter	Advantages	Disadvantages
Huong Thanh Le, Nhan Trong Tran	A Sentiment Analyzer for Informal Text in Social Media.	2018	Decision Tree, KNN, SVM, Voting Classifier for solving sentiment analysis.	Accuracy : 63.7%	It deals with repeated characters and the effect of contrast word sentence polarity.	Has not taken care of careful investigation on the use of the contrast words.
Olga Kolchyna, Th´arsis T. P. Souza1, Philip C. Treleaven and Tomaso Aste	Methodology for Twitter Sentiment Analysis	2015	Lexicon based approach, Classification using:Naive Bayes, SVM and Hybrid model i,e Naive bayes and SVM.	Accuracy : 86%	Combined lexicon and machine learning approach which improved accuracy by 5%.	They did no perform all of the pre-processing techniques properly.

QUANTITY OF WORK - HIGH LEVEL BLOCK DIAGRAM OF OUR IMPLEMENTATION



QUANTITY OF WORK – THE MAIN CODE MODULES (WHAT THEY DO)

Serial no	Code module description	Status (% complete)	What it does ?
1.	Expand Contraction in tweets in dataset	100%	Preprocessing of text
2.	Replace chat/Abbreviated words	100%	Pre-processing of text
3.	Converting hinglish words to english	100%	Pre-processing of text
4.	Remove url, punctuation, stopwords	100%	Pre-processing of text
5.	Generation of labels uisng POS Tagging	100%	Label Generation
6.	Generation of labels uisng TextBlob	100%	Label Generation
7..	Check accuracy between manually labelled and generated data.	100%	Performance Metrics

QUANTITY OF WORK – THE MAIN CODE MODULES (WHAT THEY DO)

Serial no	Code module description	Status (% complete)	What it does ?
8.	Bar Graph for labels across the review	100%	Number of positive,negative and neutral labels
9.	Histogram for distribution of polarity	100%	Distribution of polarity across the entire reviews in dataset
10.	Logistic Regression Classifier	100%	Classify the text and get the sentiments of the tweets using keras.
11.	Support Vector Machines	100%	Classify the text and gives polarity of the tweets.
12.	Random Forest Classifier	100%	Classify the text and get the sentiments of the tweets for sentiment analysis.

QUALITY OF WORK – MILESTONES THAT ARE DONE AND WORKING

Serial no	Milestone description	Status (% complete)	Comments
1.	Replacement of slang and abbreviated words from the dataset.	100%	
2.	All the preprocessing used for getting better and only important data from the tweet.	100%	
3.	Various methods to generate labels for the dataset.	100%	
4.	Three different Classifiers used by providing the pre-processed input for doing the sentiment analysis.	100%	
5.	Data Visualisation done for obtaining the distribution of polarity among all the reviews.	100%	

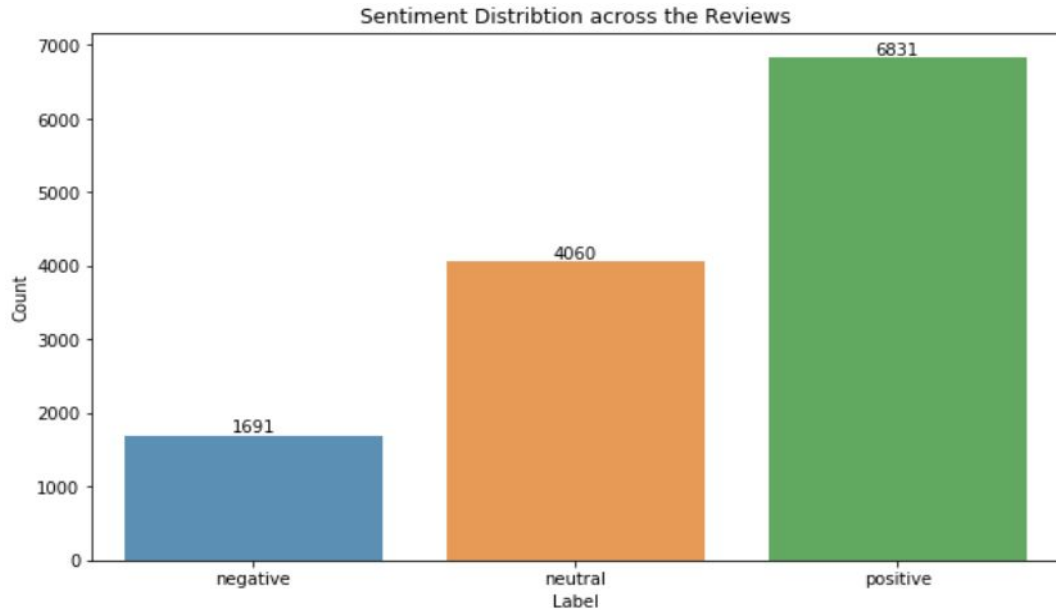
RESULTS OBTAINED

- The sentiments obtained for the reviews using TextBlob method:

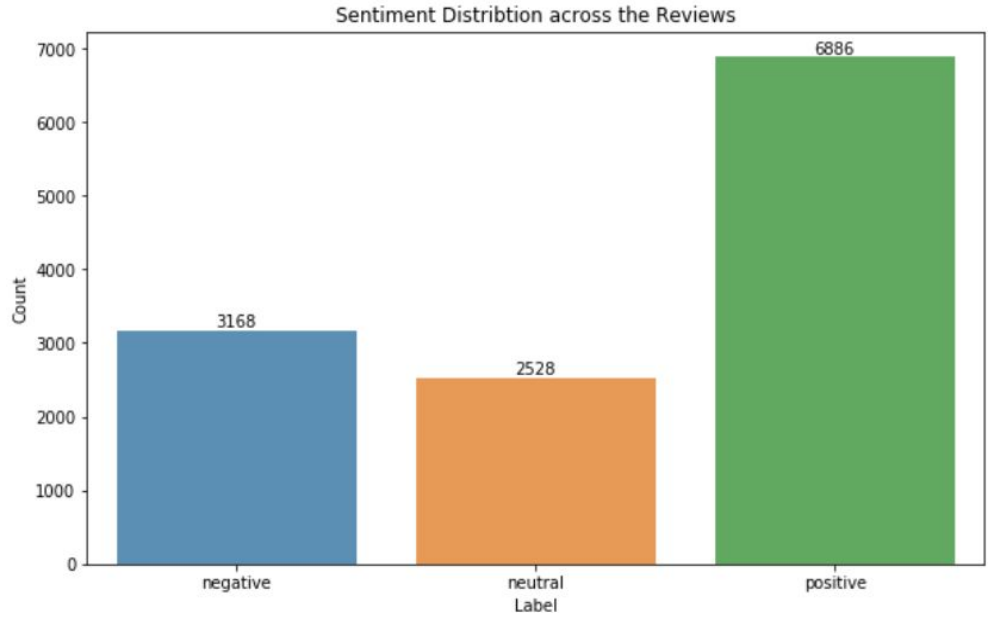
	text_raw	label
0	love kabir superb movie story character sc...	positive
1	sweet teddy bear guy chicks magnet behave like...	positive
2	obsessed bekhayali happened	negative
3	character kabir needs psychological treatment ...	negative
4	beautiful person talented actor simplicity sa...	positive

RESULTS OBTAINED

- Number of positive, negative and neutral labels across the reviews.



Using TextBlob



Using POS Tagging

RESULTS OBTAINED

Performance metrics for the three classifiers:

- ***Logistic Regression:***

Logistic Regression Accuracy Score -> 86.93579148124603

	precision	recall	f1-score	support
negative	0.51	0.81	0.62	258
neutral	0.94	0.81	0.87	1144
positive	0.91	0.91	0.91	1744
accuracy			0.87	3146
macro avg	0.79	0.85	0.80	3146
weighted avg	0.89	0.87	0.88	3146

RESULTS OBTAINED

- ***SVM***

SVM Accuracy Score -> 87.44537147397695

	precision	recall	f1-score	support
0	0.61	0.70	0.65	291
1	0.90	0.89	0.89	842
2	0.93	0.90	0.91	1384
accuracy			0.87	2517
macro avg	0.81	0.83	0.82	2517
weighted avg	0.88	0.87	0.88	2517

RESULTS OBTAINED

- ***Random Forest Classifier***

RANDOM FOREST Accuracy Score --> 0.822949777495232

	precision	recall	f1-score	support
negative	0.98	0.20	0.33	414
neutral	0.78	0.93	0.84	987
positive	0.85	0.91	0.88	1745
accuracy			0.82	3146
macro avg	0.87	0.68	0.68	3146
weighted avg	0.84	0.82	0.80	3146

OUR TOP THREE LEARNING IN THIS PROJECT

1. Learning # 1 :We learnt how important it is to replace the slang and the hinglish words in our tweets instead of removing them as they are also contributing to the sentiments expressed by the users and by replacing these words the classifiers prediction also became more accurate

2. Learning # 2 : Since our dataset had no true labels we generated labels using SentiWordNet and TextBlob, compared by testing them on test dataset containing manually labelled tweets and observed that TextBlob is more efficient in labelling.

3. Learning # 3 : While conducting the sentiment analysis on the tweets using three different classifier Multinomial Logistic Regression , SVM and Random Forest we saw that the SVM produced better performance metrics than other classifiers.

TOP CHALLENGES UNRESOLVED SO FAR

1. Issue #1: The proposed system does not detect the emotion of the words e.g "it was damn good", "i am obsessed with this song". "damn","obsessed" has been identified as negative word. Where it is not used as a negative word in the text but a figure of speech called hyperbole.

2.Issue #2: Language diversity in social media data is a key issue which is required to be taken care.

3.Issue #3: Developing a technique to perform sentiment classification that can be applicable for any data regardless of domain.

OUR GOING FORWARD PLAN (IF ANY)

1. The Hyperbole issue mentioned in the previous slide can be improved in the further by incorporating the hyperbole detection in the system.