###### **PROJECT REPORT**

###### **ON**

**SENTIMENTAL ANALYSIS ON VOICE OF MARKET**

Submitted by:

Ananya Pradhan - 1701209004

Nicky Garg - 1701209361

Drishti Anand - 1701209197

Biswan Behera - 1701209316

Shubham Mishra - 1701209063

**4thSemester** CSE (Batch : 2017-21)

Group No: CSE02

Under guidance of :

**Mr. Pradipta Kumar Pattanayak**



Department of

Computer Science & Engineering

SILICON INSTITUTE OF TECHNOLOGY

**Silicon Hills, Patia, Bhubaneswar-751024**

**CERTIFICATE**

This is to certify that Ananya Pradhan (Regd. No: 1701209004), Nicky Garg (Regd. No: 1701209361), Drishti Anand (Regd. No: 1701209197) and Biswan Behera (Regd. No: 1701209316), Shubham Mishra (Regd. No: 1701209000) have undertaken and successfully completed the project entitled **“Sentiment Analysis on Voice of Market”** under our supervision.

This work is original and is being submitted as a part of 4th Semester project for the undergraduate curriculum.

**Signature of the Guide:. ………………………………………..……………**

**Name :. ………………………….……………………………..……………….**

**Designation: ……………………………………………………………………**

Department of

Computer Science & Engineering

SILICON INSTITUTE OF TECHNOLOGY

**Silicon Hills, Patia, Bhubaneswar-751024**

**ACKNOWLEDGMENT**

The satisfaction that accompanies the successful completion of this project would be incomplete without the mention of the people who made it possible, without whose constant guidance and encouragement would have made efforts go in vain. We consider ourselves privileged to express gratitude and respect towards all those who guided us through the completion of this project.

We convey thanks to our project guide Mr. Pradipta KumarPattanayak of Computer Science and Engineering Department for providing encouragement, constant support and guidance which was of great help to complete this project successfully.

Last, but not the least, we wish to thank our parents for financing our studies in this college as well as for constantly encouraging us to learn engineering. Their personal sacrifice in providing this opportunity to learn Engineering is gratefully acknowledged.

### Ananya Pradhan Nicky Garg

Regd. No: 1701209004 Regd. No: 1701209361

### Drishti Anand Biswan Behera

Regd. No: 1701209197 Regd. No: 1701209316

**Shubham Mishra**

Regd. No: 1701209063

Department of

Computer Science & Engineering

SILICON INSTITUTE OF TECHNOLOGY

**Silicon Hills, Patia, Bhubaneswar-751024**

**STUDENT DECLARATION**

We hereby declare that the project report entitled **“SENTIMENTAL ANALYSIS ON VOICE OF MARKET”** submitted in partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering to Biju Patnaik University of Technology is our original work and not submitted to any other university or Institute for the award of any degree or diploma.

### Ananya Pradhan Nicky Garg

Regd. No: 1701209004 Regd. No: 1701209361

### Drishti Anand Biswan Behera

Regd. No: 1701209197 Regd. No: 1701209316

**Shubham Mishra**

Regd. No: 1701209063

**Department of**

Computer Science & Engineering

SILICON INSTITUTE OF TECHNOLOGY

**Silicon Hills, Patia, Bhubaneswar-751024**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter** | **Contents** | **Page** |
| **1** | **Introduction and Statement of Problem** |  |
|  | 1.1 : Introduction | 5 |
|  | 1.2 : Statement of Problem | 5 |
|  | 1.3 : Organization of Report | 6 |
| **2** | **Review of Related Work** | 6 |
| **3** | **Algorithm & Implementation** | 15 |
| **4** | **Conclusion and Result** |  |
|  | 4.1 : Snapshots | 21 |
|  | 4.2 : Scope for Future Work | 22 |
| **5** | **References** | 23 |
| **6** | **Appendix** |  |
|  | I: List of Figures | 50 |
|  | II: List of Tables | 55 |
|  | III: Web Link | 60 |

**ABSTRACT**

In this project, we investigate the utility of linguistic features for detecting the sentiment of Twitter messages. We evaluate the usefulness of existing lexical resources as well as features that capture information about the informal and creative language used in micro-blogging. We take a supervised approach to the problem, but leverage existing hash tags, emoticons, re-tweets, and text analysis in the Twitter data for building training set.

There are wide applications of Twitter Sentiment Analysis like Online Commerce, Voice of Market, Voice of Customer, Government, etc.

We choose to focus on the Twitter Sentiment Analysis on Voice of Market (VOM). Voice of the Market is about determining what customers are feeling about products or services of competitors. Accurate and timely information from the Voice of the Market helps in gaining competitive advantage and new product development. Detection of such information as early as possible helps in direct and target key marketing campaigns.

Sentiment Analysis helps corporate to get customer opinion in real-time. This real-time information helps them to design new marketing strategies, improve product features and can predict chances of product failure.

1. **INTRODUCTION AND STATEMENT OF PROBLEM:**

## Introduction

Sentiment Analysis is the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic or product, etc. is positive, negative, or neutral.

Our topic revolves around the main sub topicknown as “Voice of Market”. It mainly focuses on a multinational American Corporation-Nike. Nike during its 30​thanniversary made​Colin Kaepernick as its brand ambassador. During that time Kaepernick took a major step bynot standing during the national anthem in one of his games. His intention was not to disrespect the country but to provide justice to the constitution as a protester of Police brutality that was persisting due to racism.

**1.1.1. Purpose:**

The purpose of this document is to give a detailed Description of Twitter Sentiment Analysis of Voice of Market. It will explain the System Constraints, background and foreground and how this program interacts with the user.

**1.1.2. System Overview:**

* OS: Linux/Windows 10
* RAM: 1GB
* Physical Memory: 512GB/2TB
* Compiler: Python3.7

**1.2 Problem Statement**

Twitter is a popular social networking website where members create and interact with messages known as “tweets”. This serves as a mean for individuals to express their thoughts or feelings about different subjects. Various different parties such as consumers and marketers have done sentiment analysis on such tweets to gather insights into products or to conduct market analysis. Furthermore, with the recent advancements in machine learning algorithms, we are able improve the accuracy of our sentiment analysis predictions.

In this report, we will attempt to conduct sentiment analysis on “tweets” using various different machine learning algorithms. We attempt to classify the polarity of the tweet where it is either positive or negative. If the tweet has both positive and negative elements, the more dominant sentiment should be picked as the final label.

We use the dataset from [Kaggle](https://www.kaggle.com/eliasdabbas/5000-justdoit-tweets-dataset) which was crawled from Twitter API. The data provided comes with emoticons, usernames and hashtags which are required to be processed and converted into standard form. We also need to extract useful features from the text such unigrams and bigrams which is a form of representation of the “tweet”.

## We use various machine learning algorithms to conduct sentiment analysis using the extracted features. However, just relying on individual models did not give a high accuracy so we pick the top few models to generate a model ensemble. Ensembling is a form of meta learning algorithm technique where we combine different classifiers in order to improve the prediction accuracy. Finally, we report our experimental results and findings at the end.

## 1.3 Organization of report

## 1.3.1 Data Collection

Data in the form of raw tweets is retrieved by using the *API tweepy* which used for real time twitter streaming API. The API requires us to register a developer account with Twitter and fill in parameters such as consumerKey, consumerSecret, accessTokenaccess, and TokenSecret. This API allows to get all random tweets or filter data by using keywords. Filters supports to retrieve tweets which match a specific criterion defined by the developers.

The dataset we took, is already available on Kaggle. This was crawled during the time of the controversy. After fetching the dataset, it is stored as a csv file.

**1.3.2 Data Pre-processing**

It is a data-mining technique that is used to convert raw data into a format that has improved semantics. It involves Data Cleaning i.e., removal of URLs, Hashtags, Commas, Full-stops,

Hyphens, semi-colons, etc.

It also involves removal of stop-words, Stop words are defined as the most commonly used words in sentences which carry very little meaning such as “and”, “then” , “the”, ”is” etc.

The objective of removing stop words is to reduce the dimension of the data set and make it easier for the machine to classify. In some cases the use of stop words in data set may mislead the tweet classifier to categorize a particular tweet in the wrong category.

Stop words also include commonly used phrases, prepositions, conjuctions,

determiners, adjectives etc.

**1.3.3 Classifier**

We can use using the Naive Bayes classifier and SVM (Support Vector Machines) for the project.

The Naive Bayes algorithm is an intuitive method that uses the probabilities of each attribute belonging to each class to make a prediction. It is the supervised learning approach you would come up with if you wanted to model a predictive modelling problem probabilistically.

Naive Bayes simplifies the calculation of probabilities by assuming that the probability of each attribute belonging to a given class value is independent of all other attributes. This is a strong assumption but results in a fast and effective method.

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification challenges.  In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular co-ordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well.

1. **REVIEW OF RELATED WORK:**

Applying sentiment analysis on Twitter is the upcoming trend with researchers recognizing the scientific trials and its potential applications. The challenges unique to this problem area are largely attributed to the dominantly informal tone of the micro blogging. Pak and Paroubek [5] rationale the use microblogging and more particularly Twitter as a corpus for sentiment analysis.

1. [**Go, Bhayani and Huang (2009)**](http://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf)

They classify Tweets for a query term into negative or positive sentiment. They collect training dataset automatically from Twitter. To collect positive and negative tweets, they query twitter for happy and sad emoticons.

They try various features – unigrams, bigrams and Part-of-Speech and train their classifier on various machine learning algorithms – Naive Bayes, Maximum Entropy and Scalable Vector Machines and compare it against a baseline classifier by counting the number of positive and negative words from a publicly available corpus. They report that Bigrams alone and Part-of-Speech Tagging are not helpful and that Naive Bayes Classifier gives the best results.

1. [**Pak and Paroubek (2010)**](https://pdfs.semanticscholar.org/ad8a/7f620a57478ff70045f97abc7aec9687ccbd.pdf)

They identify that use of informal and creative language make sentiment analysis of tweets a rather different task . They leverage previous work done in hashtags and sentiment analysis to build their classifier. They use Edinburgh Twitter corpus to find out most frequent hashtags. They manually classify these hashtags and use them to in turn classify the tweets. Apart from using n-grams and Part-of-Speech features, they also build a feature set from already existing MPQA subjectivity lexicon and Internet Lingo Dictionary. They report that the best results are seen with n-gram features with lexicon features, while using Part-of-Speech features causes a drop in accuracy.

1. [**Koulompis, Wilson and Moore (2011)**](http://www.aclweb.org/website/old_anthology/S/S13/S13-2.pdf#page=526)

They investigated the utility of linguistic features for detecting the sentiment of Twitter messages. They evaluated the usefulness of existing lexical resources as well as features that capture information about the informal and creative language used in microblogging. They took a supervised approach to the problem, but leverage existing hashtags in the Twitter data for building training data.

1. [**Saif, He and Alani (2012)**](http://oro.open.ac.uk/34929/1/76490497.pdf)

They discuss a semantic based approach to identify the entity being discussed in a tweet, like a person, organization etc. They also demonstrate that removal of stop words is not a necessary step and may have undesirable effect on the classifier.

All of the aforementioned techniques rely on n-gram features. we improve our results by using more basic techniques used in Sentiment Analysis, like stemming, two-step classification and negation detection and scope of negation.

Negation detection is a technique that has often been studied in sentiment analysis. Due to presence of such words, the meaning of nearby words becomes opposite. Such words are said to be in the scope of negation. Many researches have worked on detecting the scope of negation.

1. **Algorithm & Implementation:**

We use different feature sets and machine learning classifiers to determine the best combination for sentiment analysis of twitter. We also experiment with various pre-processing steps like - punctuations, emoticons, twitter specific terms and stemming. We investigated the following features - unigrams, bigrams, trigrams and negation detection. We finally train our classifier using various machine-learning algorithms - Naive Bayes, Decision Trees, Maximum Entropy and Support Vector Machines.

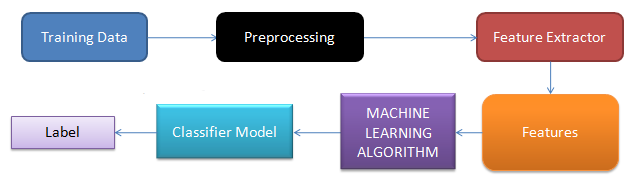


FIG 1 : SCHEMATIC BLOCK REPRESENTATION OF THE METHODOLOGY

We use a modularized approach with feature extractor and classification algorithm as two independent components. This enables us to experiment with different options for each component.

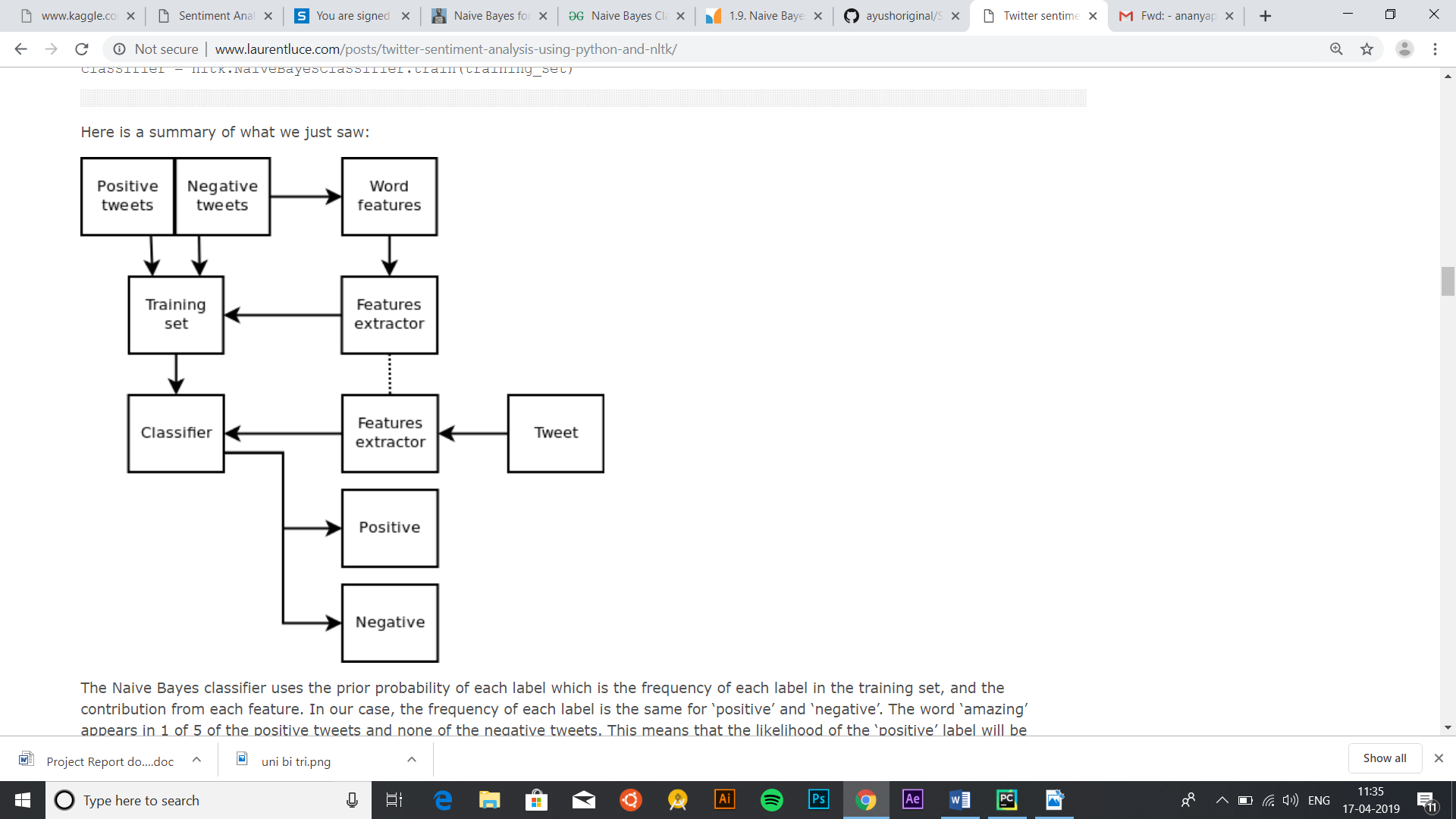


FIG 2 : OUR APPROACH

**3.1 DATASETS**

The major challenge for text classification and sentiment analysis is to get a labelled data set. However, if we do not have one we have to manually label a dataset.

|  |  |  |
| --- | --- | --- |
| Class | Count | Example |
| neg | 529 | I’m not a loyal Nike Customer. #TakeAKnee |
| neu | 3770 | There are some amazingly hilarious news on my newsfeed. #nike #collin |
| pos | 483 | Best marketing move they ever could have done 🙇🏽‍♂️ #Justdoit https://t.co/sKSREo5AOd |

**3.2 PRE-PROCESSING**

Raw tweets scraped from twitter generally result in a noisy dataset. This is due to the casual and carefree nature of people’s usage of social media. Tweets have certain special characteristics such as retweets, emoticons, user mentions, hashtags, etc. which have to be suitably extracted. Therefore, raw twitter data has to be normalized to create a dataset which can be easily learned by various classifiers. We have applied an extensive number of pre-processing steps to standardize the dataset and reduce

its size.

We first do some general pre-processing on tweets which is as follows.

* Lower Case - Convert the tweets to lower case.
* URLs - I don't intend to follow the short urls and determine the content of the site, so we can eliminate all of these URLs via regular expression matching or replace with generic word URL.
* @username - we can eliminate "@username" via regex matching or replace it with generic word AT\_USER.
* #hashtag - hash tags can give us some useful information, so it is useful to replace them with the exact same word without the hash. E.g. #nike replaced with 'nike'.
* Punctuations and additional white spaces - remove punctuation at the start and ending of the tweets. E.g.: ' the day is beautiful! ' replaced with 'the day is beautiful'. It is also helpful to replace multiple whitespaces with a single whitespace.

We handle special twitter features as follows.

**3.3FEATURE EXTRACTION**

We extract two types of features from our dataset, namely unigrams and bigrams. We create a frequency distribution of the unigrams and bigrams present in the dataset and choose top N unigrams and bigrams for our analysis.

**3.3.1 Unigrams**

Probably the simplest and the most commonly used features for text classification is the presence of single words or tokens in the text. We extract single words from the training dataset and create a frequency distribution of these words. Total unique words are extracted from the dataset. Out of these words, most of the words at end of frequency spectrum are noise and occur very few times to influence classification. We, therefore, only use top N words from these to create our vocabulary for sparse vector. The frequency of a word is inversely proportional to its rank in the frequency table.

Raw misses Swimming Class. http://plurk.com/p/12nt0b

Normalized misses swimming class URL

Raw - @98PXYRochester HEYYYYYYYYY!! its Fer from Chile again

Normalized - heyy its fer from chile again

Raw - Sometimes, You gotta hate #Windows updates.

Normalized sometimes you gotta hate windows updates

Raw @Santiago\_Stephhii come talk to me i got candy :)

Normalized - hii come talk to me i got candy EMO\_POS

Example tweets from data set with their normalised versions. (To see this please refer to the snapshots attached.)

**3.3.2 Bigrams**

Bigrams are word pairs in the dataset which occur in succession in the corpus. These features area good way to model negation in natural language like in the phrase – This is not good. A total of1954953 unique bigrams were extracted from the dataset. Out of these, most of the bigrams at endof frequency spectrum are noise and occur very few times to influence classification.

**3.3.3 N-grams**

N-gram refers to an n-long sequence of words. Probabilistic Language Models based on Unigrams, Bigrams and Trigrams can be successfully used to predict the next word given a current context of words. In the domain of sentiment analysis, the performance of N-grams is unclear.

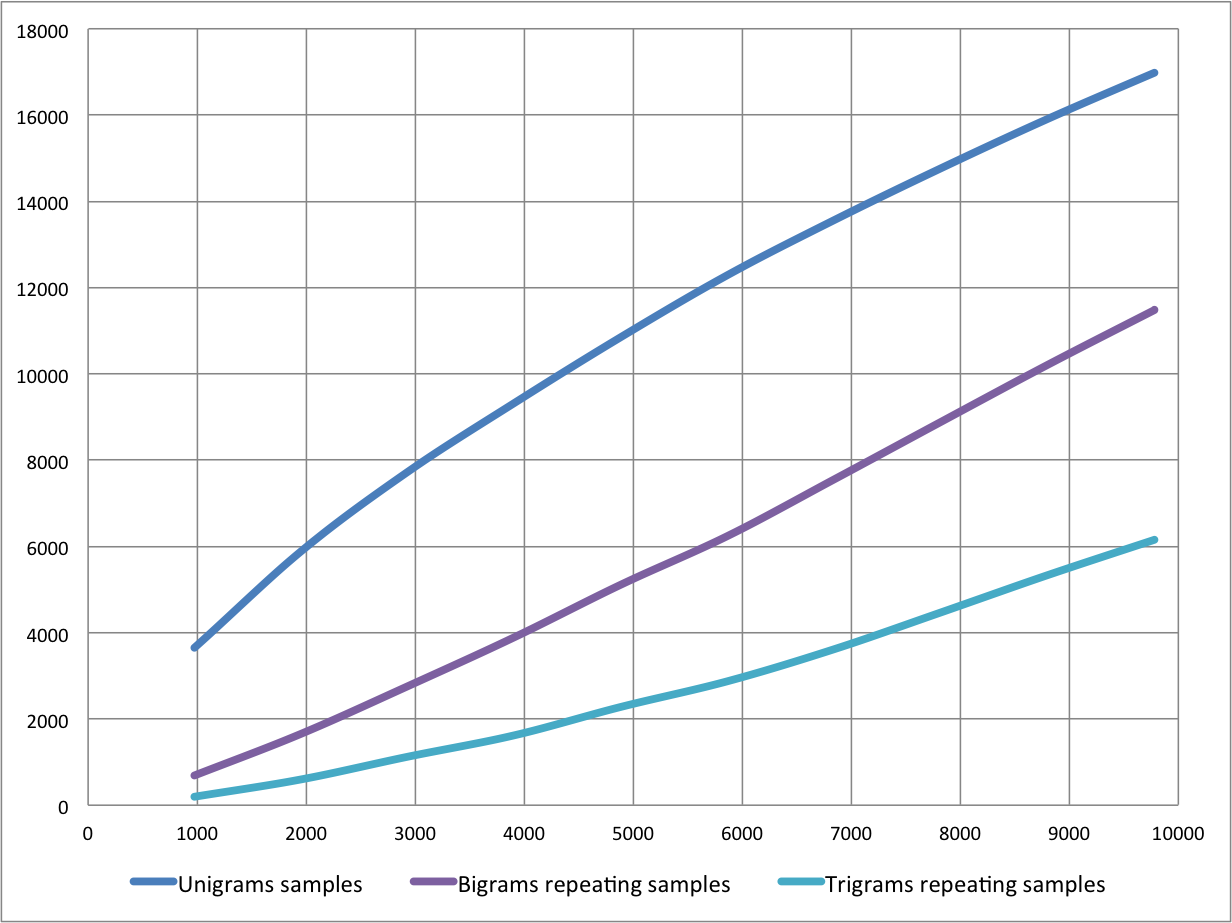


FIG 3:NO. OF REPEATING SAMPLES

**3.4 FEATURE REPRESENTATION**

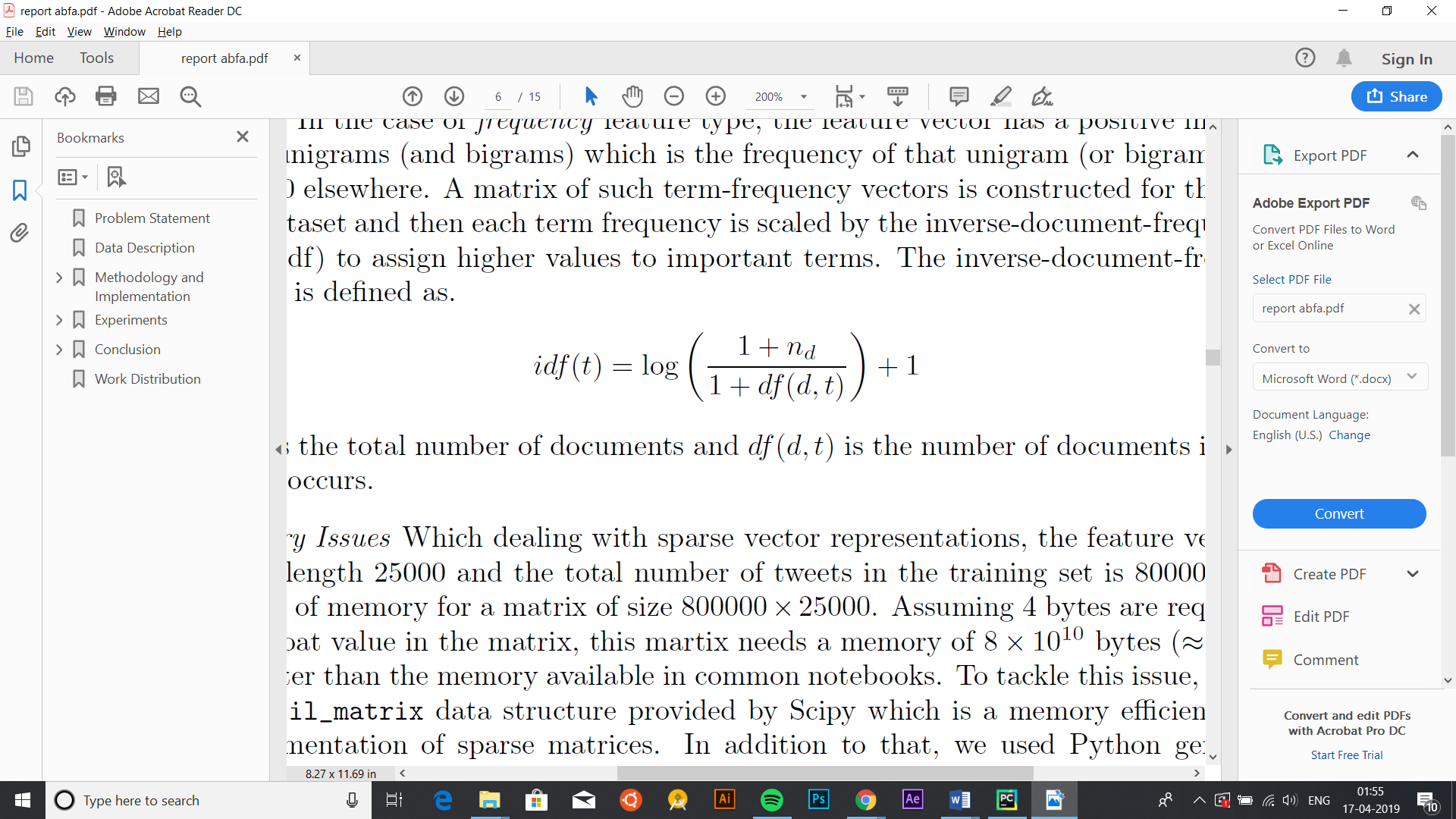
After extracting the unigrams and bigrams, we represent each tweet as a feature vector in either sparse vector representation or dense vector representation depending on the classification method.

**3.4.1 Sparse Vector Representation**

Depending on whether or not we are using bigram features, the sparse vector representation of each tweet is either of length 15000 (when considering only unigrams) or 25000 (when considering unigrams and bigrams). Each unigram (and bigram) is given a unique index depending on its rank.

The feature vector for a tweet has a positive value at the indices of unigrams (and bigrams) which are present in that tweet and zero elsewhere which is why the vector is sparse. The positive value at the indices of unigrams (and bigrams) depends on the feature type we specify which is one of presence and frequency.

* ***presence*** In the case of presence feature type, the feature vector has a 1 at indices of unigrams (and bigrams) present in a tweet and 0 elsewhere.
* ***frequency*** In the case of frequency feature type, the feature vector has a positive integer at indices of unigrams (and bigrams) which is the frequency of that unigram (or bigram) in the tweet and 0 elsewhere. A matrix of such term-frequency vectors is constructed for the entire training dataset and then each term frequency is scaled by the inverse-document-frequency of the term (idf) to assign higher values to important terms. The inverse-document-frequency of a term t is defined as.



Where nd is the total number of documents and df(d; t) is the number of documents in which the term t occurs.

**3.3.2 Dense Vector Representation**

For dense vector representation we use a vocabulary of unigrams of size 90000 i.e. the top 90000words in the dataset. We assign an integer index to each word depending on its rank (starting from1) which means that the most common word is assigned the number 1, the second most common word is assigned the number 2 and so on. Each

tweet is then represented by a vector of these indices which is a dense vector.

**3.4 CLASSIFIERS**

After a lot of detailed research on the various types of Classifier Algorithms that could be applied to this project, we concluded with two of the most commonly used classifiers – The Naïve Bayes Classifier and SVM(Support Vector Machines).

* 1. **Naïve-Bayes’ Classifier:**
     1. **Introduction:**

It is a classification technique based on Bayes’ Theorem with an assumption of independence among predictors. In simple terms, a Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, consider the dataset with 3 features, Temperature, Humidity, Outlook and Windy. The temperature being ‘Hot’ has nothing to do with the humidity or the outlook being ‘Rainy’ has no effect on the winds. Hence, the features are assumed to be **independent**., all of these properties independently contribute to the probability and that is why it is known as ‘Naïve’.

The fundamental Naive Bayes assumption is that each feature makes an:

* Independent – no pair of features are dependent.
* Equal – Each Vector is given the same weight (or importance)

contribution to the outcome.

Naïve Bayes model is easy to build and particularly useful for very large data sets.

Along with simplicity, Naïve Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

Above,

* *P*(*c|x*) is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
* *P*(*c*) is the prior probability of *class*.
* *P*(*x|c*) is the likelihood which is the probability of *predictor* given *class*.
* *P*(*x*) is the prior probability of *predictor*.

**3.1.2. How Naïve Bayes algorithm works?**

Let’s understand it using an example. Below I have a training data set of weather and corresponding target variable ‘Play’ (suggesting possibilities of playing). Now, we need to classify whether players will play or not based on weather condition. Let’s follow the below steps to perform it.

Step 1: Convert the data set into a frequency table

Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.

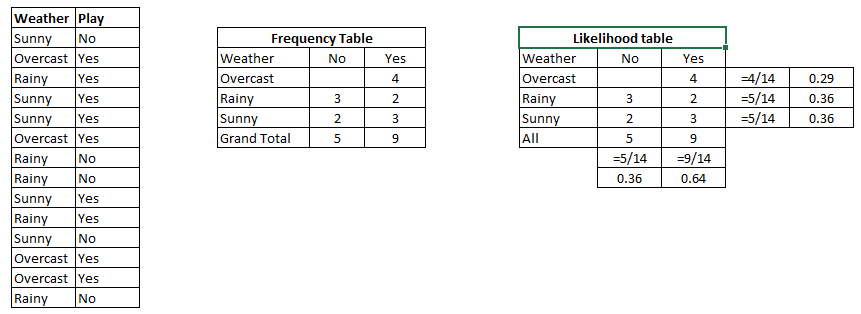
[](https://www.analyticsvidhya.com/wp-content/uploads/2015/08/Bayes_41.png)

TABLE 1: NAÏVE BAYES CLASSIFICATION

Step 3: Now, use Naïve Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

**Problem:**Players will play if weather is sunny. Is this statement is correct?

We can solve it using above discussed method of posterior probability.

P(Yes | Sunny) = P( Sunny | Yes) \* P(Yes) / P (Sunny)

Here we have P (Sunny |Yes) = 3/9 = 0.33, P(Sunny) = 5/14 = 0.36, P( Yes)= 9/14 = 0.64

Now, P (Yes | Sunny) = 0.33 \* 0.64 / 0.36 = 0.60, which has higher probability.

Naïve Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

**3.1.3. Applications of Naïve Bayes Algorithms:**

* Real time Prediction
* Multi class Prediction
* Text classification/ Spam Filtering/ Sentiment Analysis
* Recommendation System
  1. **SVM Classifier (Support Vector Machine):**

**3.2.1 What is Support Vector Machine?**

“Support Vector Machine” (SVM) is a supervised [machine learning algorithm](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2?utm_source=blog&utm_medium=understandingsupportvectormachinearticle) which can be used for both classification or regression challenges. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the hyper-plane that differentiate the two classes very well.

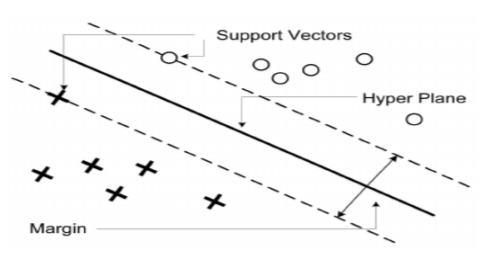


FIG 4: SVM CLASSIFICATION

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

**Linear case:**

The dataset is represented as , | Where is dataset of n rows that consisting of element , . This dataset is a pair of and where having a dimensional feature vector and having label of classes. Decision function that separates two classes is defined as



Where ‘w’ is normal vector and is the intercept of the hyperplane. and specifies the position of the hyperplane. The objective of hyperplane to search the optimal separating hyperplane. Mathematically SVM problem can be formulated as





The optimal hyperplane can be found by solving above linearly constrained quadratic optimization equation. On solving this equation, we get following solution:



**Non-Linear case:**

For non-linear classification, SVM transforms the original data into a higher dimension. Then it seeks the linear optimal separating hyperplane or decision boundary in a a new dimension. The decision function is given as

****is non-linear mapping function.

**3.2.2Experimental Setup and Result Analysis**

The proposed algorithm is implemented in Python with NLTK . About 3745 tweets of people about three Indian politicians Narender Modi, Rahul Gandhi and Arwind Kejriwal are collected. Each tweet is labelled as positive class or negative class. Two types of Support Vector Machine are applied for classifications of twitter data. Eighty percent data are used for training and twenty percent data are used for testing.Table 1 shows the types of support vector machines.

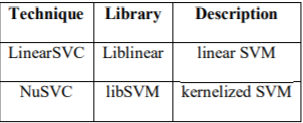
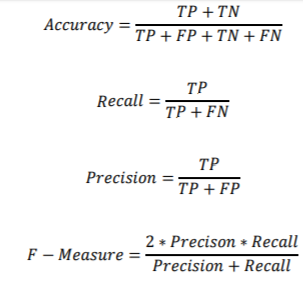
****

FIG 5: SUPPORT VECTOR MACHINE

The performance of the proposed algorithm is analyzed using four parameters i.e. Accuracy, Recall, Precision, and F-measure.



Here, True positive (TP) defines the number of positive tweets that are correctly classified, as positive, whereas false positive (FP) is the number of negative tweets that are incorrectly classified as positive. True negative (TN) is the number of negative instances that are correctly classified as negative and false negative (FN) is the number of positive tuples that are incorrectly classified as negative tweets.

**3.2.3 Results**

Figure 6 shows positive and negative sentiments or views of public towards three Indian politicians. Different leaders have different sentiment results according to their working procedure. Among these leaders, Narender Modi is more successful politician

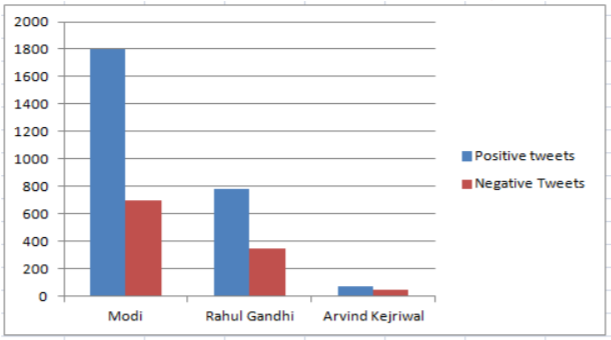
****

FIG 6: SENTIMENT ANALYSIS OF OPINIONS OF PEOPLE

Two types of SVM are applied for classification of tweets. Table 2 shows the experimental results of linear SVM and Kernel SVM. Figure 4 shows the Performance Comparison between Linear SVM and Kernel SVM.

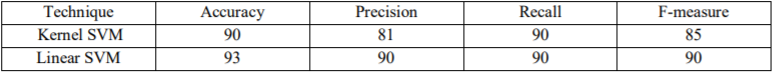


TABLE 2: EXPERIMENTAL RESULT OF LINEAR SVM AND KERNEL SVM

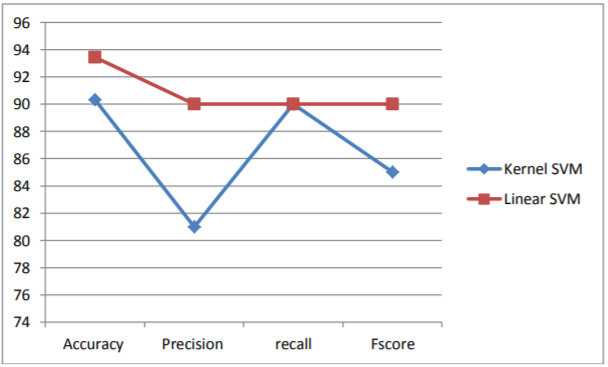


FIG 7: PERFORMANCE COMPARISION BETWEEN LINEAR SVM AND KERNEL SVM

It has been analysed that the Linear SVM gives more accuracy as compared to Kernel SVM. The proposed approach is compared to other sentiment analysis technique that is based on the unsupervised learning approach .The unsupervised learning approach gives 78.6% accuracy whereas our approach gives above 80% accuracy.

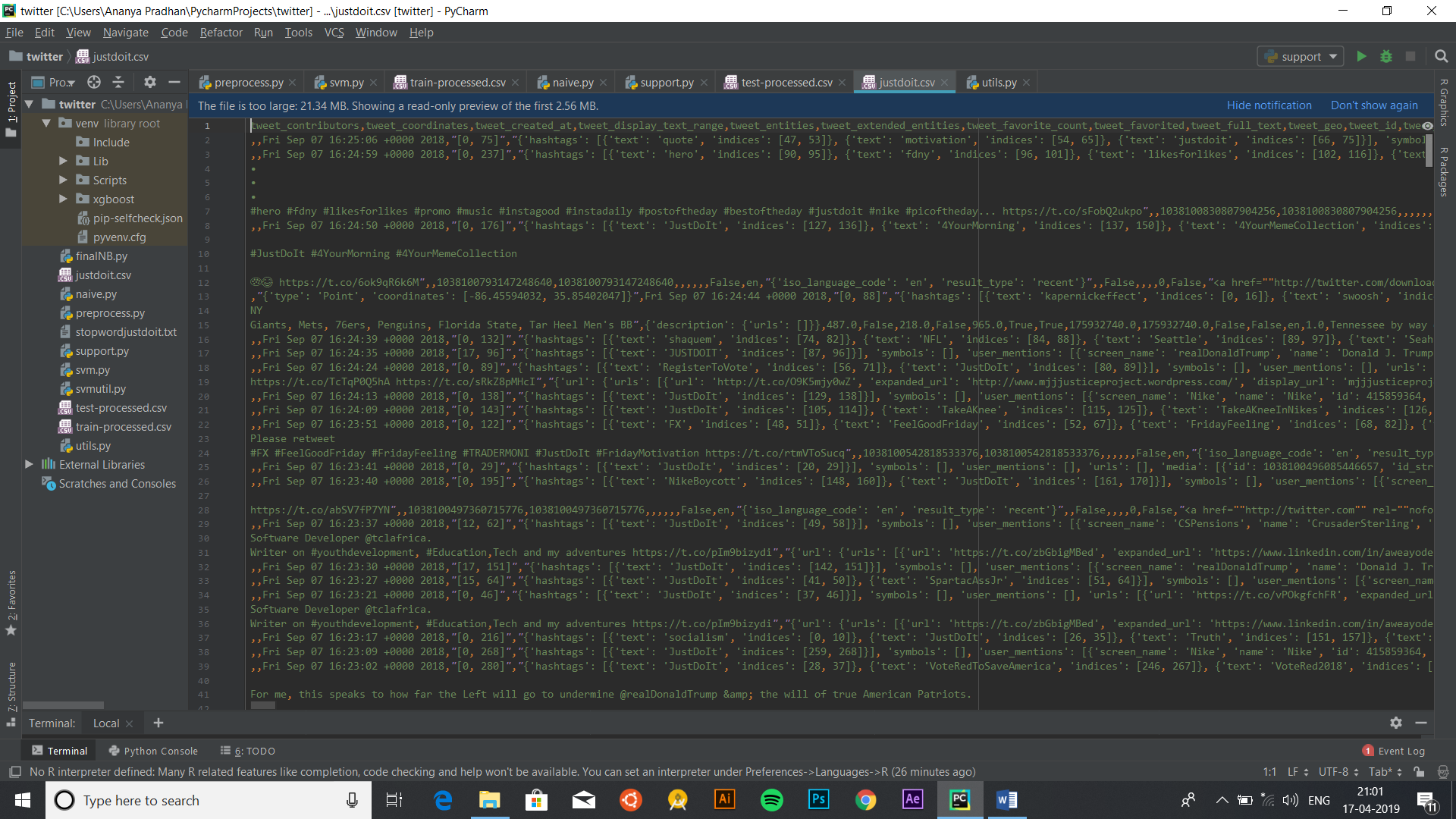
1. **Conclusion and Result:**

Sentiment analysis has been an important tool for brands looking to learn more about how their customers are thinking and feeling. It is a relatively simplistic form of analytics that helps brands find key areas of weakness (negative sentiments) and strengths (positive sentiments). Moving forward, sentiment analysis is finding a place in other organizations. During Brexit and the 2016 US election, these data tools were used to measure emotions and attempt to predict the outcome of these events. This has led to non-brand organizations turning to sentiment analysis for their own needs. Additionally, the insights gained from these tools are becoming much deeper, as a result of emerging social media platforms and features.

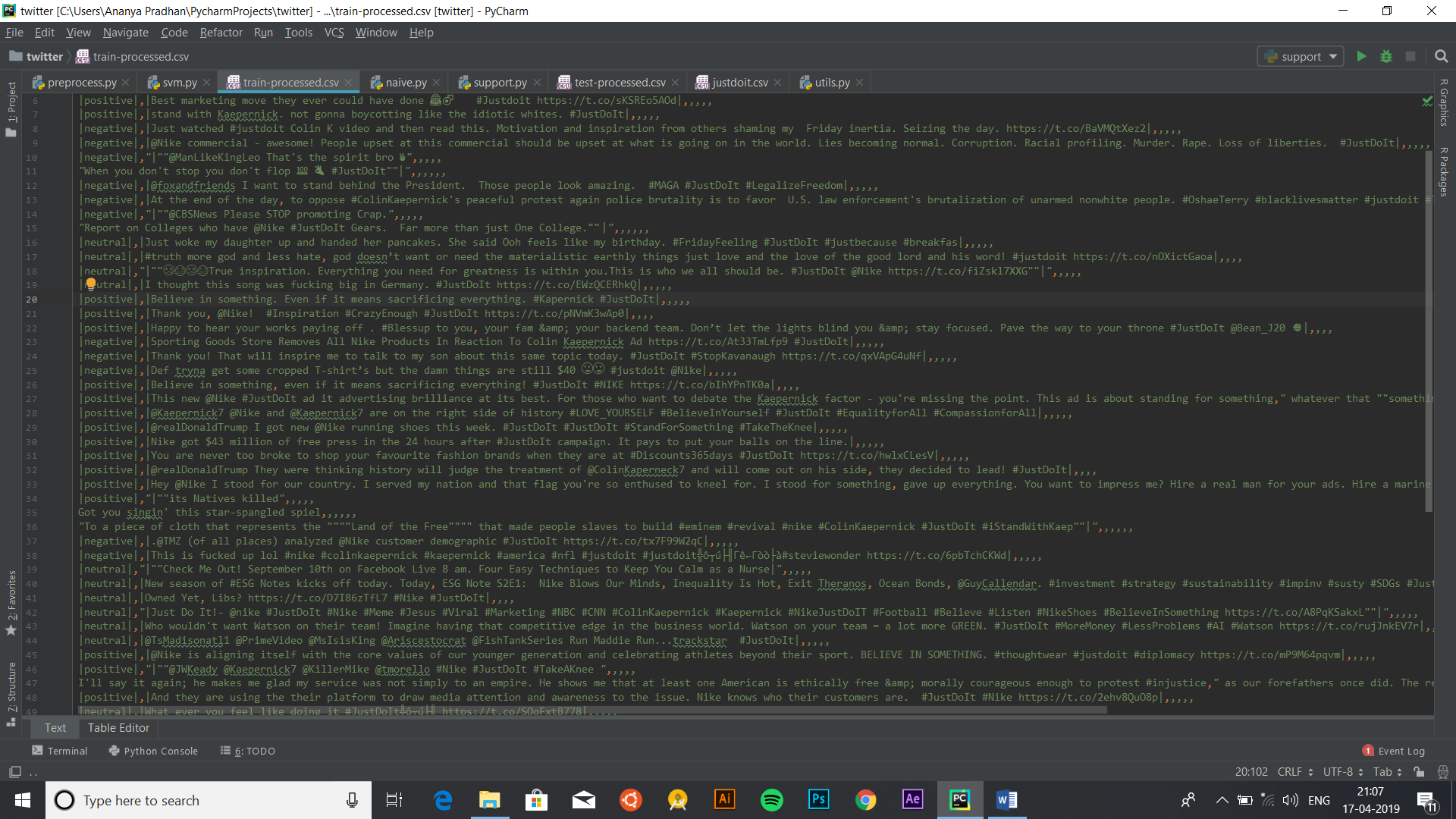
Despite all the challenges and potential problems that threatens Sentiment analysis, one cannot ignore the value that it adds to the industry. Because Sentiment analysis bases its results on factors that are so inherently humane, it is bound to become one the major drivers of many business decisions in future. Improved accuracy and consistency in text mining techniques can help overcome some current problems faced in Sentiment analysis. Looking ahead, what we can see is a true social **democracy** that will be created using Sentiment analysis, where we can harness the wisdom of the crowd rather than a select few “experts”. A democracy where every opinion counts and every sentiment affects decision making.

With more and more organizations turning to sentiment analysis to measure and predict outcomes, as well as better understand consumer behaviors, these tools are quickly building a reputation that is going to help propel it forward into the future and towards deeper and more accurate conclusions and insights.

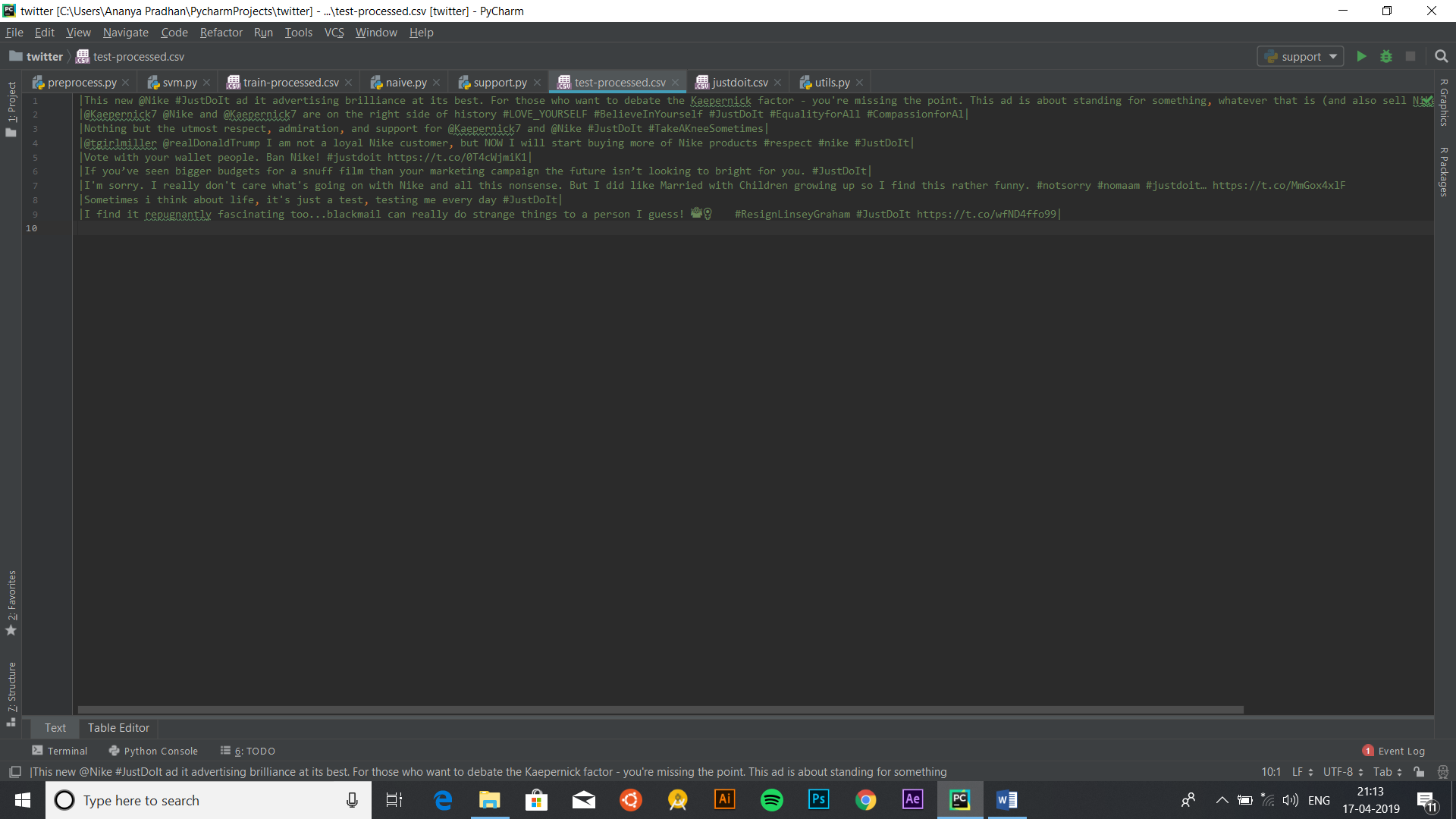
**4.1 Snapshots**



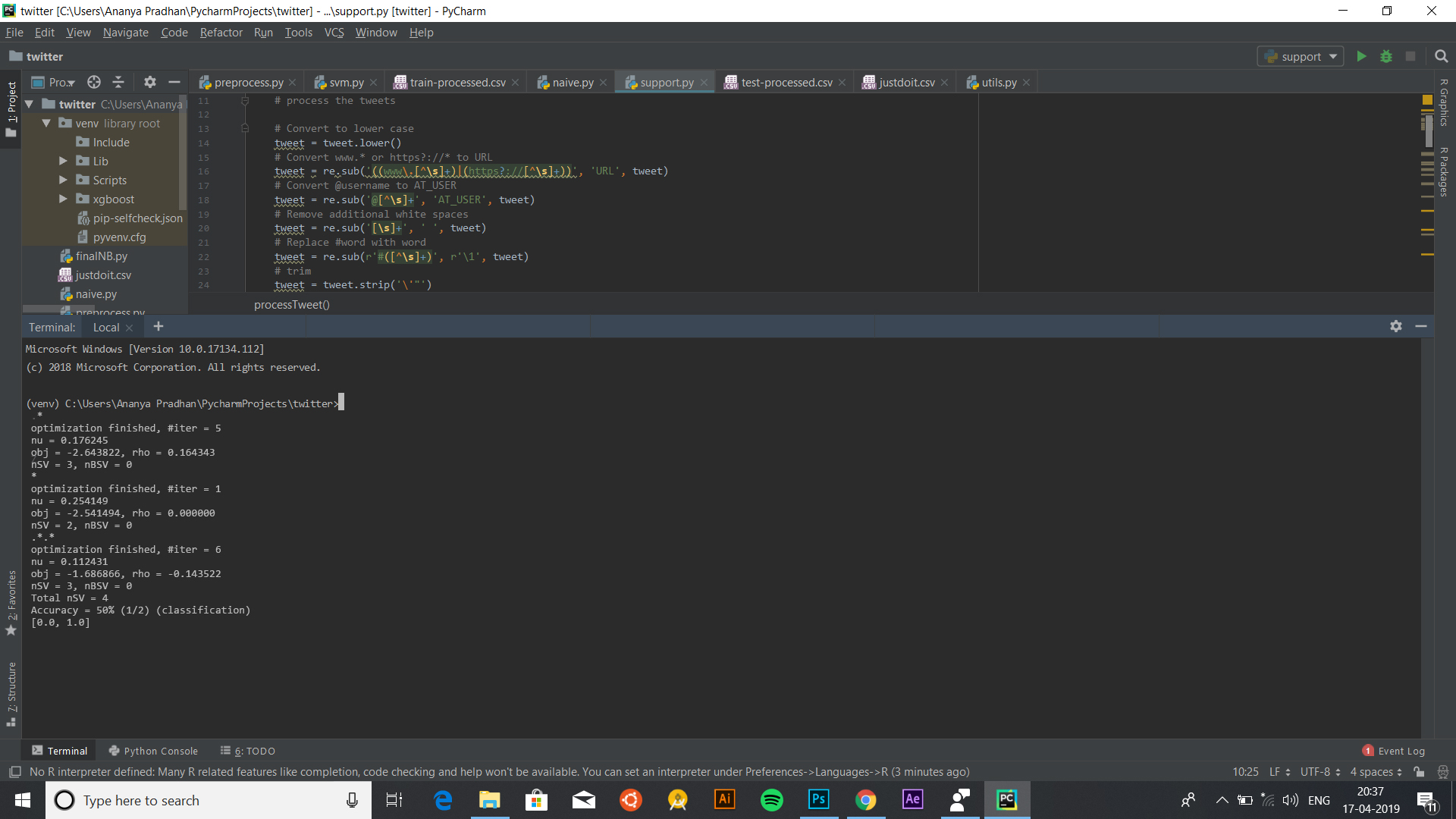
NOISY DATA SET



PROCESSED TRAINING SET

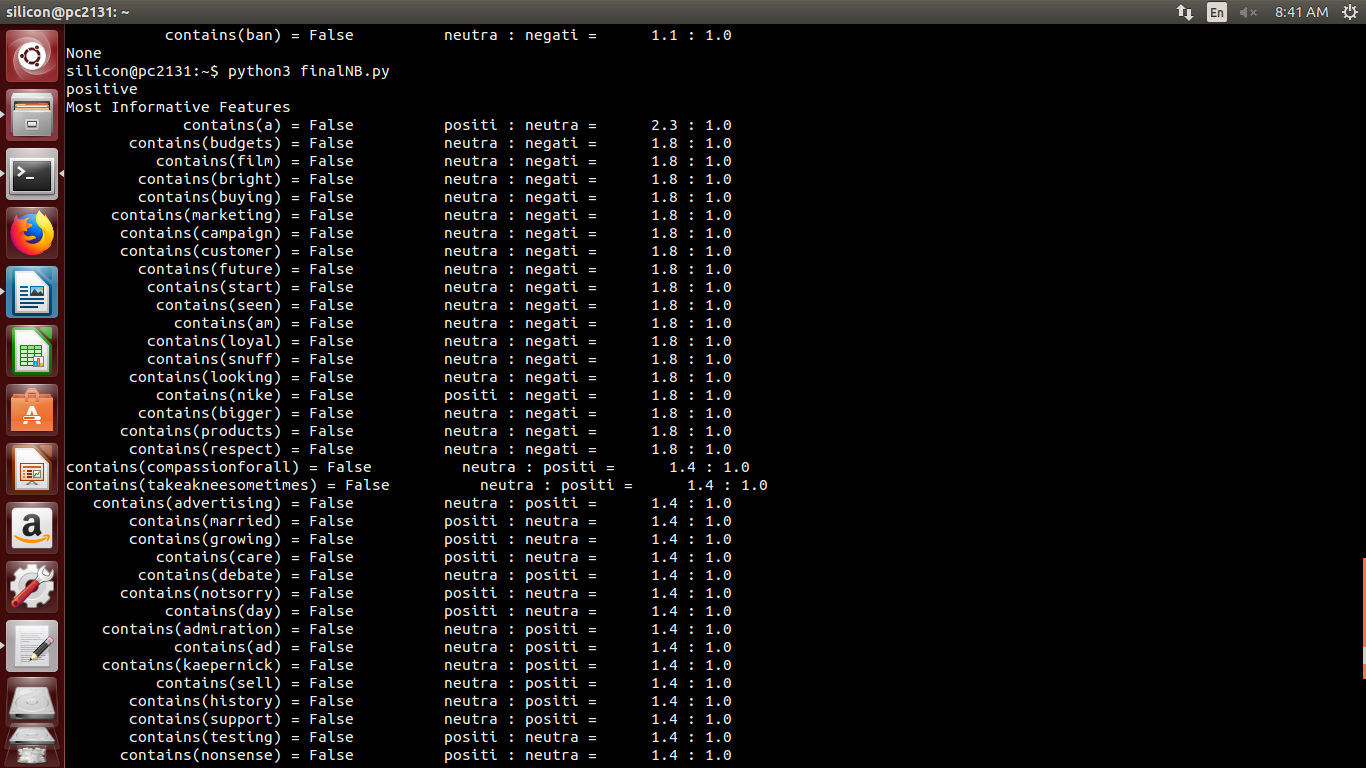


PROCESSED TEST SET

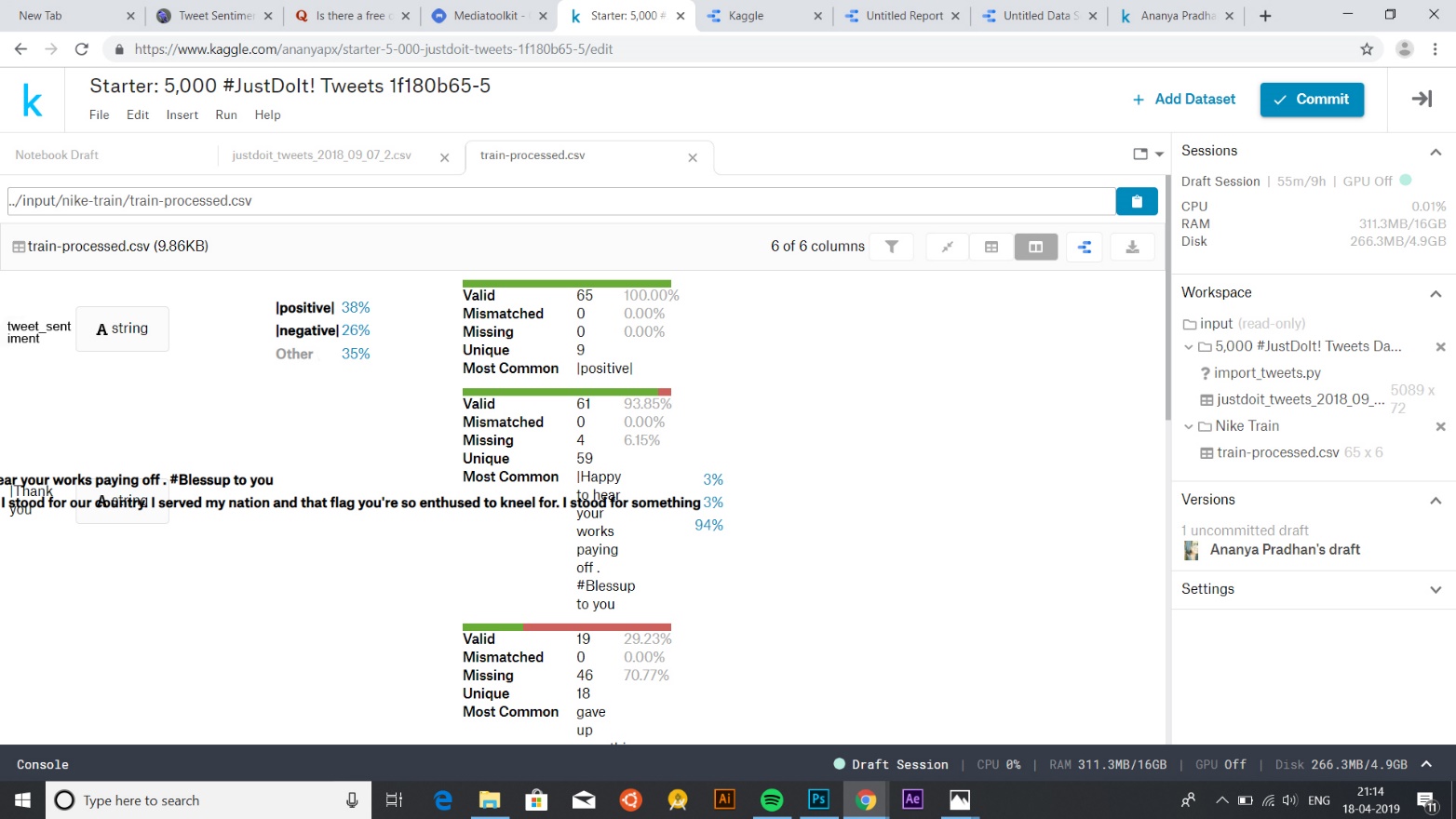


SVM OUTPUT

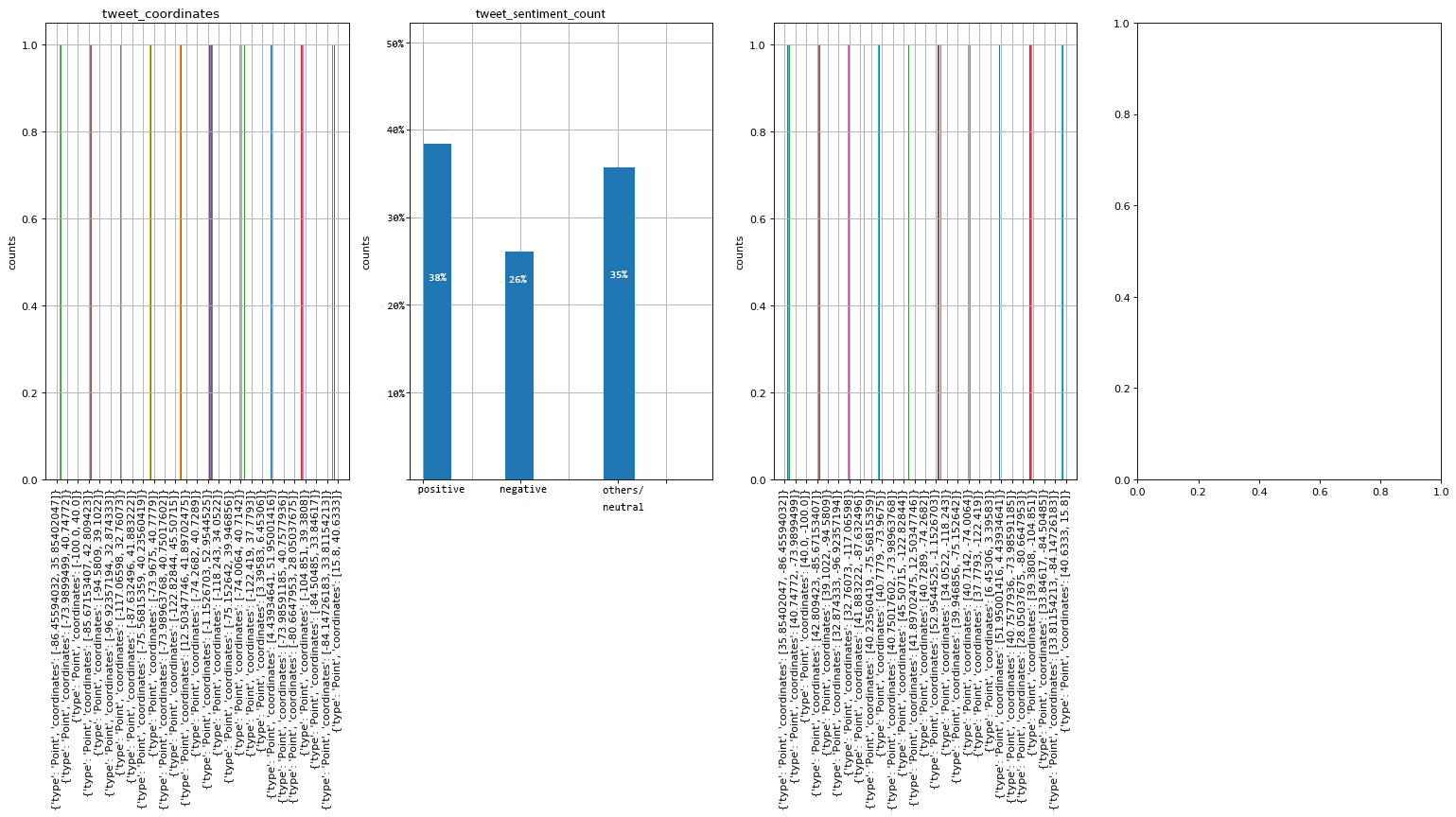
­­­­

****

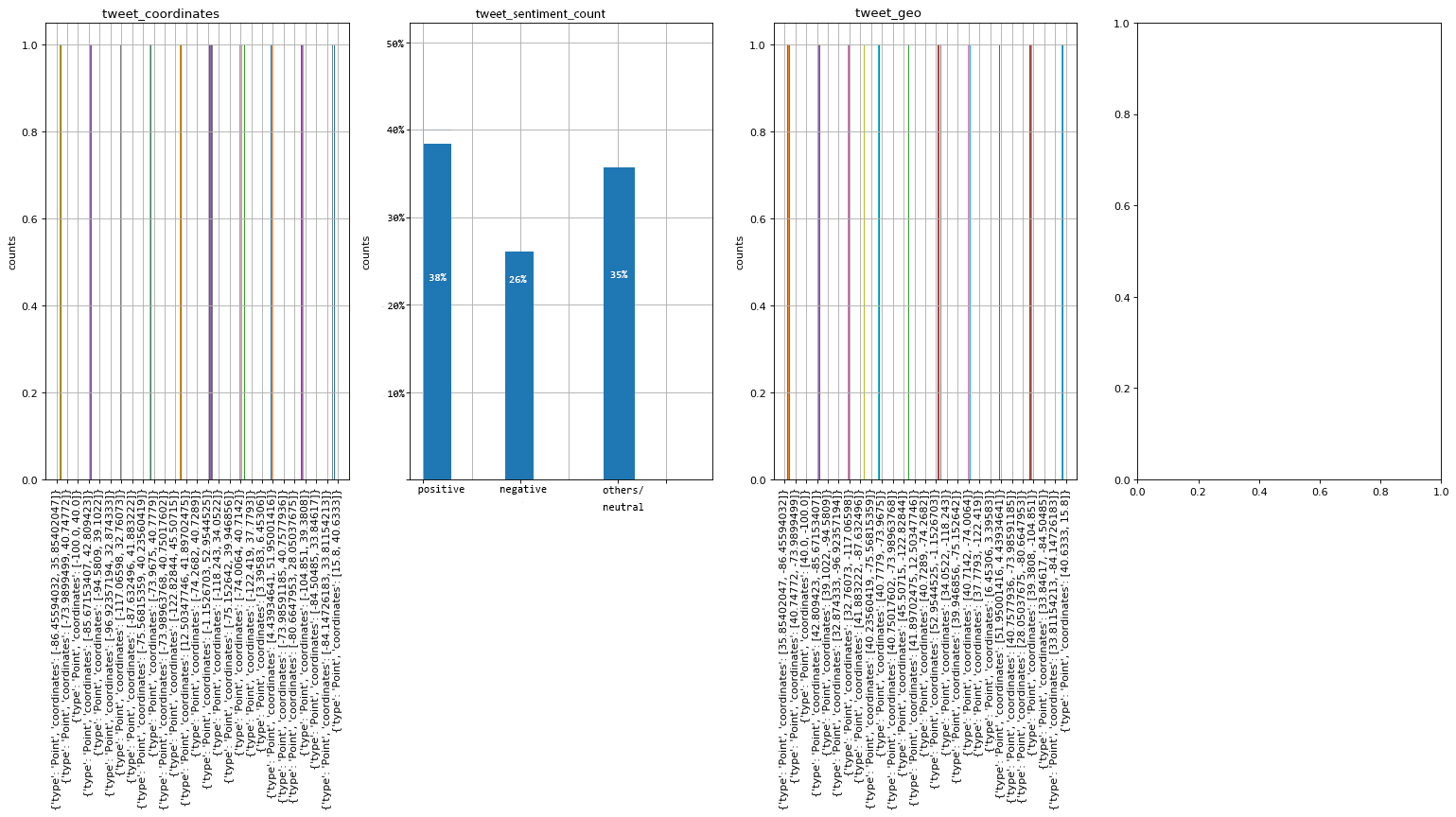
NAIVE BAYES OUTPUT AND LIST OF MOST INFORMATIVE FEATURES



PERCENTAGES OF SENTIMENTS ON KAGGLE LEADERBOARD ON OUR ALREADY CLASSIFIED RESULTS



HISTOGRAM OF OUR CLASSIFIED RESULTS ON KAGGLE



HERE, I CODED A HISTOGRAM DISPLAY ON KAGGLE NOTEBOOK AS A PART OF MY DRAFT ON KAGGLE WHICH DISPLAYS THE :

a) Tweet coordinates b) The sentiment analysis of the tweets c) the geo location of the tweet

**4.2 Scope for future work:**

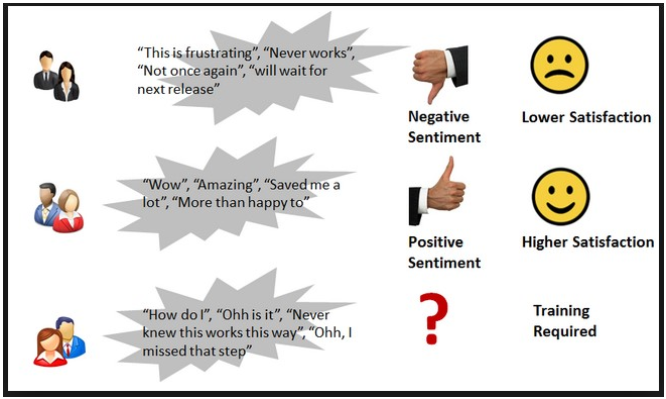
Sentiment Analysis has been more than just a social analytic tool. It is a field that is still being studied, although not at great lengths due to the intricacy of this analysis. The ability to understand sarcasm, hyperbole, positive feelings, or negative feelings has been difficult, for machines that lack feelings. Algorithms have not been able to predict with more than 60% accuracy the feelings portrayed by people. Yet with so many limitations this is one field which is growing at great pace within many industries. Companies want to accommodate the sentiment analysis tools into areas of customer feedback, marketing, CRM, and ecommerce.

****

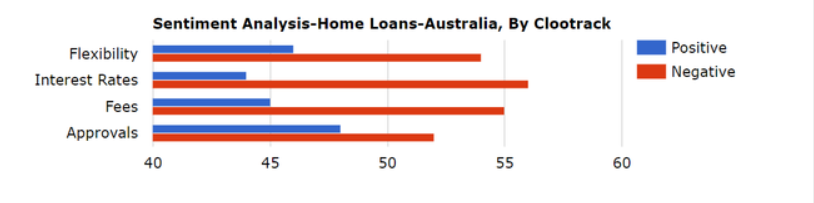
Sentiment analysis methods till now have been used to detect the polarity in the thoughts and opinions of all the users that access social media. Researchers and Businesses are very interested to understand the thoughts of people and how they respond to everything happening around them. Companies use this to evaluate their advertisement campaigns and to improve their products.

Lexicon sentiment creation is labour intensive and there are already unsupervised methods to create them. Such algorithms will also have to understand and analyse natural text concept-wise and context-wise. Collecting opinions on the web will still requires processing that can filter out un-opinionated user-generated content and also to test the trustworthiness of the opinion and its source.

A sentiment analysis tool Tweview had predicted the winner of the show X factor but eventually that person came second. So improvements on the analysis is one scope which is under way by many tools available on the web.

****

Sarcasm is the biggest challenge that sentiment analysis faces. Machine or algorithms with no emotion will find it extremely difficult to differentiate when users are commenting sarcastically.



The above graph shows the result of analysis based on customer reviews and conversations online in the particular segment. The source data used for the analysis are user conversations available publicly online like public forums and social media.

**The future of Sentiment Analysis are:**

## Deeper, Broader Insights from Sentiment Analysis

## Greater Personalization for Audiences

## Not Just For Marketers and Brands

## Algorithm-Based Sentiment Analysis Plateaus

1. **References:**

<https://www.kaggle.com/eliasdabbas/5000-justdoit-tweets-dataset>

<http://www.scoop.it/t/social-media-monitoring-tools-and-solutions>

<http://www.brandwatch.com/2013/12/social-data-gets-the-x-factor/>

<https://www.linkedin.com/pulse/future-sentiment-analysis-shahbaz-anwar>

<http://ijcsma.com/publications/october2017/V5I1014.pdf>

[Naïve Bayes Classifier](https://www.geeksforgeeks.org/naive-bayes-classifiers/) – Geeks for Geeks.

Jason Brownlie in [Understanding Machine Learning Algorithms](https://machinelearningmastery.com/naive-bayes-for-machine-learning/)

1. **Appendix**

**6.I List of Figures**

Figure 1: Schematic block representation of the methodology.

Figure 2: Our approach for sentiment analysis.

Figure 3: Number of repeating samples.

Figure 4: SVM Classification

Figure 5: Support Vector Machine

Figure 6: Sentiment analysis of opinions of people

Figure 7: Performance comparision between linear SVM and kernel SVM.

**6.2 List of Tables**

TABLE 1 : Naïve Bayes Classification

TABLE 2: Experimental result of Linear and Kernel SVM

**6.3 Web Links**

1. This project can be found on our github repository: