Comparative Study on Machine Learning Techniques based Sentiment- Analysis of Textual Documents from Twitter

Sayantan Bhattacharya, Mahmadul Hasan, Md Rahmat Ali, Fahad Ghayas, Chandra Das and Shilpi Bose

***Abstract*--WWW has rapidly developed into a cutting-edge platform for people to voice their views and ideas on various subjects, trends, and problems. Thecontents created by individualsin different platforms including online forums, chatting platforms, blogs, etc. serve an important role in decision-making. Advertising, opinion polls, online surveys, market forecast, corporate information, social media discussions, etc. are the primary sources of content creation. Sentiment analysis deals with the issue of extracting sentiments from text data and classifying the author's viewpoint on a specific entity into no more than three predetermined categories: positive, negative, and neutral. Still now there exists any article which elaborates step by step procedure using Python in a user friendly manner. In this regard, in this article, we describe the step by step sentiment analysis procedure to categorize Twitter's highly unstructured data (as a case study) in a very user friendly approach using Python and also we compare the performance of different machine learning techniques discussed here. The conclusion of this evaluated study reflects the effective and usefulness of different machine learning techniques for sentiment analysis.**

***Index Terms*--Sentiment Analysis, Machine Learning, Classification, Feature Selection, Python.**

I. INTRODUCTION

INtoday's world of social media everyone reflects their image using different platforms like twitter, Facebook, Instagram etc [1, 2]. There has been constant effort and development to make these platforms better for the users and developers. Various technique, algorithm have been used to add new features to make it more attractive, presentable and reliable. The use of micro-blogging platforms like twitter has increased significantly during the last several years. Because of this expansion, businesses and other media outlets are seeking for new methods to glean information from Twitter about what customers think and feel about their products and services.

Our project is based on sentimental analysis of raw twitter data. As data is the new gold, today in the era of social media we are very much surrounded by people's opinion having different perspective which is now everyone's looking at, so all the company are willing to invest time, manpower and money to extract and use that data for these purposes. Our model will provide three types of output based on the given input data, as an illustration, an organisation can assess a marketing campaign's effectiveness or learn how to change it for greater growth success by getting consumer feedback on it.

Twitter is used as a platform for sharing opinions on anything. As opinion is subjective in nature, it needs to be calculated and then summarized. Among these platforms, twitter is very popular [3,4]. Twitter was established in 2006 and is now the most widely used micro-blogging platform worldwide. It enables users to follow one another and submit short messages that are strictly limited to 280 characters long, known as Tweet. Twitter welcomed third-party developers from the start, providing a flexible application programming interface (API), and it also experiences an unheard-of level of celebrity popularity. Twitter offers a great platform for promoting and selling businesses. Twitter supports strong communities. Companies use Twitter as a platform to monitor consumer sentiment towards both their own brand and that of their rivals in almost real time. Perhaps, Twitter has recently been used most successfully, for breaking news and understanding impact of news on our society. So, extracting information from Twitter data is a very important task now a day.

In addition to identifying and forecasting product roadmap effectiveness, sentiment analysis can also tell customer success whether customer service, usage of the product, minor bugs, has its strength or weakness. So, sentiment analysis [1-8] of Twitter data is very much important. Still now no work is done where step by step method for sentiment analysis using Python is given. In this perspective, in our work we have presented a step by step sentiment analysis model using Python as case study for sentiment analysis of Twitter data to compare classification of sentiments using different machine learning models based on raw twitter data. The paper is

This work was a final year project inthe Department of CSE, Netaji Subhash Engineering College.

Sayantan Bhattacharyais with Department of CSE, Netaji Subhash Engineering College. (e-mail: sayantanofficialmail@gmail.com).

Mahmadul Hasan is with Department of CSE, Netaji Subhash Engineering College. (e-mail: mahmadulhasancse2019@nsec.ac.in)

Md Rahmat Ali is with Department of CSE, Netaji Subhash Engineering College. (e-mail: mdrahatalicse2019@nsec.ac.in)

Fahad Ghayas is with Department of CSE, Netaji Subhash Engineering College. (e-mail:fahadghayascse2019@gmail.com)

Chandra Das is with Department of CSE, Netaji Subhash Engineering College. (e-mail: chandra.das@nsec.ac.in)

Shilpi Bose is with Department of CSE, Netaji Subhash Engineering College. (e-mail: shilpi.bose@nsec.ac.in)

Volume 2, Issue 2
https:// doi.org/10.15864/ajac.22004

organized in the following format: Section 2 gives a brief overview of sentiment analysis techniques while in section3 the step by step procedure is given. In section 4, we analyze the results while in section 5 conclusions are given.

II. System Overview

A brief overview of sentiment analysis is given below[1]:

Steps of sentiments analysis:

A brief overview of different steps for sentiment analysis are given below:

- **Preprocessing :**People frequently express their feelings and sentiments on social media in effortless ways. Due to the high degree of unstructuredness in the data collected from these social media platforms' posts, audits, comments, remarks, and criticisms, sentiment and emotion analysis by computers is challenging. Pre-processing is therefore a crucial step in the data cleaning process because it has a significant effect on many approaches that come after it. Tokenization, stop word removal, POS tagging, and other pre-processing operations are necessary for a dataset's structure. Action must be taken since some of these pre-processing techniques run the risk of erasing crucial data for sentiment and expression analysis.
- **Feature extraction :**The preprocessed collection contains many distinctive qualities. Using the feature extraction method, we extract the features from the processed dataset. This feature is later utilised to compute the positive and negative polarity of a sentence using models like unigram and bigram, which is useful for gauging people's attitudes. The popular feature extraction techniques are: (1) word vectorization or word embedding (2) 'Bag of Words' (BOW) (3) N-gram method (4) TFIDF method (5) deep learning based methods (most popular are Word2Vec, GloVeetc).
- **Different techniques for Sentiment analysis and emotion detection:**The techniques for sentiment analysis and emotion detection can be generally categorised as lexicon-based, machine learning-based, and deep learning-based approaches. To address the shortcomings of both approaches, the hybrid strategy combines statistical and machine learning methods. Another division of machine learning called "transfer learning" enables the application of previously learned models to new domains that are related.

Sentiment analysis tasks:

The difficult interdisciplinary job of sentiment analysis can be broken down into the subsequent tasks. These are given below:

- **Subjectivity Classification:** Subjectivity classification is the process of identifying whether or not a sentence expresses an opinion.
- **Sentiment classification:** A sentence's polarity, or whether it expresses a positive or negative view, must be determined once we have established if it is opinionated.
- **Complimentary Tasks:** Finding information or opinion-makers is what it is. Finding the source of an idea can be done directly or indirectly.

Levels of sentiment analysis:

The tasks of sentiment analysis can be carried out at various levels of granularity. These are:

- **Document level:** In the document level the sentiment is extracted from the entire review provided by the user and a whole opinion is classified based on the overall sentiment of the opinion giver. The target here is to classify a tweet as negative, positive and neutral. Example "Shifali got a pen a few days ago. It is such a nice pen, although a little expensive. The pen quality is awesome too. I simply love it!" This review will be negative or positive, if classified? This task works best when the tweet is written by an individual and expresses an sentiment or review on a single entity.
- **Sentence level:** Typically, there are two parts to this process: A statement is divided into objective and subjective categories called subjective division. Positive and negative emotions are used to categorize subjective statements.
- **Feature/Aspect level:** In order to assess if an opinion is good, negative, or neutral, it is necessary to identify and extract object properties that have been commented on by the opinion holder. Synonyms for features are categorized, and a feature-based summary of numerous reviews is created.

III. Proposed Section

Here we have presented the detailed approach of sentiment analysis about the topic “farmer’s protest happened in India” on tweeter in Fig. 1. First, the Twitter data has been collected and is pre-processed. Second, from the clean text features are extracted using one of the feature selection techniques. Thirdly, to create a training set, a part of the data is classified as either positive or negative Tweets using a method. The labeled training set and the extracted features are then used classify or label the remaining data through the trained classifier. Finally, we applied machine learning algorithms to predict the tweets and the basic plot the performance metrics. The following subsections go into great detail about each of the processing stages.

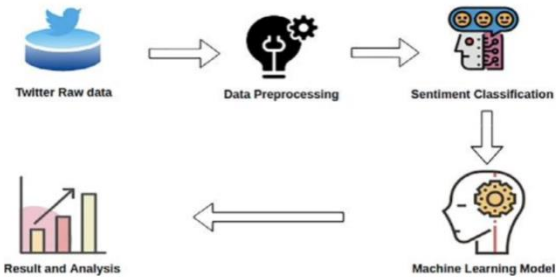

Fig.1: Sentiment Analysis Step by Step Approach

Dataset Collection:
We have gathered the raw data using Tweepy API, an open-source Python framework that uses consumer keys and private access tokens for authentication. For instance, since the farmers' protest began around November 2020, we chose March 31, 2021, as the beginning date and November 1, 2020, as the conclusion date. A customised script that was created to specifically retrieve tweets each day and use them to store them in a Python list included the Date Time module. For instance, while conducting a search using the term "farmers protest," all tweets including the phrases "protest," "farmers," and "farmers' protest" were grouped together. The tweets were collected from distinct people and will be kept in a CSV file.

Data preprocessing:
We need our data to be organized in such a way that it can be understood by the machine. For that we need to preprocess the data in multi-step process. It is important to preprocess data, which necessitates the use of a method called data cleaning, which entails converting raw data into machine-readable code. Since we use large amounts of tweets as raw data, we need to clean it to remove discrepancies to avoid inconsistent data. We will first check for duplicate and null data in dataset and remove it using. dropna() and drop duplicates. Then we remove @-mentions, retweets, special characters and symbols using sub() or replace. Tokenization is the process of breaking up large amounts of text into tokens. It is a crucial stage in the modeling of text data. To lessen the inflection towards their root forms, we apply a porter stemmer. This was accomplished by adding a prefix, suffix, or a typical morphological or inflexional ending to the nouns. The fully pre-processed data is finally saved in a new pandas column in our data frame called Cleaned Tweets.

Bag of words:
Different discrete characteristics can be found in the pre-processed dataset. Adjectives, verbs, and nouns are extracted as part of feature extraction techniques, and these aspects are later used to determine the overall polarity of the sentence. In ur model we have used Bag of Words model which is a very popular feature extraction technique from a text.

Lexicon based Tweet data labeling :
The lexicon-based approach was utilized in our project to label every tweet data. This approach has the advantage of being easy to comprehend and can be modified by humans without difficulty. Using this technique, we can capture the semantic orientation of the tweet and label it as either neutral, positive, or negative. Sentiment analysis retrieves subjectivity and polarity from text, while semantic orientation measures the polarity and strength of the text. In our case, adjectives and adverbs were used to expose the semantic orientation of the tweets. The sentiment orientation value was then computed by combining the adverbs and adjectives. We used the Python package TextBlob to label each tweet. TextBlob gives each tweet a unique score and then returns two values: polarity and subjectivity. Polarity is characterised by a value between 1 and 1, where 1 denotes a negative emotion and 1 denotes a good emotion. Negative words can flip the polarity, bringing it below zero. Subjectivity, which varies from 0 to 1, describes how much factual information and personal opinion are included in a tweet. High subjectivity suggests more personal opinion. The sentiment score is assigned based on the polarity value, scores below zero indicate negative sentiment, scores above zero indicate positive sentiment, and a score of zero indicates neutral attitude. The count of the number of sentiments was then analyzed, revealing that a large number of individuals had neutral feelings about the protest, indicating that they neither support the farmers' protest nor the Government. The tweet data was then divided into two sets, one for training and the other for testing.

Sentiment Classification via Model building:
After labeling the training data, we gave it to four well-known supervised machine learning algorithms for training, including Naive Bayes, Decision Tree, Random Forest, and Support Vector Machine. We then applied them to test data to assess their performance.

Naïve Bayes:
The Nave Bayes classifier is a straightforward probabilistic classifier that performs categorization using the idea of mixture models. The mixture model is predicated on the idea that each of the specified classes is a part of the mixture. The mixture model's components signify the likelihood that any term will adhere to a specific component. They are also referred to as "generative classifiers" as a result. The Nave Bayes classifier determines the maximum likelihood of any word to fall into a specific category using Bayes Theorem.

Decision tree:
A decision tree is a type of tree-structured classifier used to represent decision rules and outcomes based on dataset attributes. The tree is made up of leaf nodes, which reflect the decision's outcome, branches, which represent the decision rules, and internal nodes, which represent the qualities. The leaf nodes show if a sentiment is good, negative, or neutral whereas the decision nodes are used to make judgments and include several branches. The dataset is first regarded as the root node, or starting point, to gather information in the instance of tweets. The algorithm employed in Decision Trees is called entropy, and it determines how the Decision Tree splits the outcomes. The Decision Tree's boundary setting is affected by the entropy parameter, whose range is 0–1.

Random forest:
A supervised machine learning algorithm known as random forest consists of a collection of decision trees. The algorithm consists of two main stages: constructing the random forest and using it to make predictions. To construct the forest, a subset of "K" features is selected at random from a total of "m" features, where "X" is much smaller than "Y". The node "A" is then calculated using the best split point of the "X" functions, and the network is divided into daughter nodes based on the optimal split point. This process is repeated until a desired number of nodes is reached. Finally, the algorithm builds a forest by repeating the first five steps "Z" times, creating "n" trees. The purpose of using the random forest

algorithm is to evaluate whether its precision is superior to that of the decision tree algorithm.

Support vector machine:
A non-probabilistic conditional linear classifier called a support vector machine (SVM) is used to categorize incoming examples into one of two groups based on a series of labeled training examples. The best line or hyperplane in two or more dimensions will be found using the SVM training method to help classify the space. The biggest margin, or the greatest separation between data points, in both forms of data is used to locate the hyperplane. We may utilize the sci-kit-learn machine learning toolkit, which is based in Python, to implement SVMs and fit the algorithms to our dataset. The methods for visualizing and analyzing the outcomes of the prediction process are also covered in this section.

IV. Results & Analysis

Here we have classified sentiment of every tweet using different machine learning classification algorithms. For evaluation we have used 80:20 splitting of training testing data. Accuracy, Precision, Recall, and F1-score are typically used as four indicators to gauge the effectiveness of sentiment classification methods. The confusion matrix shown below is used to calculate these indicators:

#	Predicted Positives	Predicted Negatives
Actual Positive Cases	Number of True Positive Cases (TP)	Number of False Negative Cases (FN)
Actual Negative Cases	Number of False Positive Cases (FP)	Number of True Negative Cases (TN)

These indicators are defined below:

$$Accuracy = \frac{TN + TP}{TN + TP + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

System Requirements: (Software/Hardware)

Software:
Google Colaboratory
Python Library & API used:
Tweepy
Numpy
Pandas
nltk (Natural Language Toolkit)
textblob
Scikit-learn

A plot of **polarity vs subjectivity** for all the tweets (on farmers' condition post Budget 2023 in India) is represented by the following graph in Fig. 2. Here polarity is mostly concentrated in the centre and subjectivity is spread all over the graph.

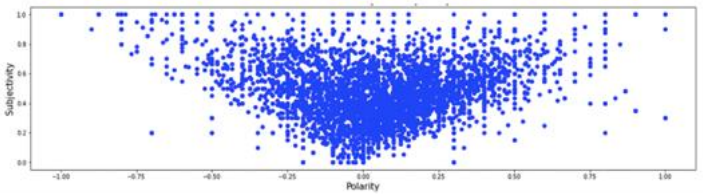

Fig. 2: polarity vs subjectivity graph of tweets on farmers' condition during post Budget 2023 in India

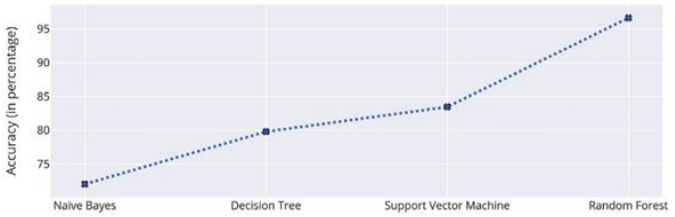

Fig. 3: classification accuracy of different classifiers

Here the classification accuracy of different classifiers is shown in the above graph in Fig.3 with respect to 80:20 training-testing splitting of twitter data. Naive Bayes algorithm showed here minimum accuracy at 72% while Random Forest algorithm has the highest accuracy of 96.6%. Accuracy level of Decision Tree and SVM are 79.8% and 83.5% respectively. The Confusion Matrices are shown here for Random Forest and SVM classifiers in Fig. 4.

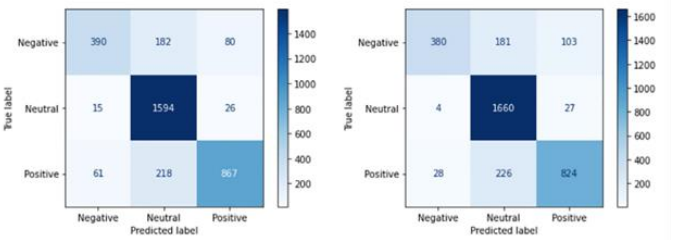

Fig. 4: Confusion Matrices are shown here for Random Forest and SVM classifiers

V. Future Scope of Up Gradation

There are many features that can be used in sentiment analysis, and the most effective features may vary depending on the specific domain or application. However, here are some examples of new features that can be considered:

Emojis and emoticons: Emojis and emoticons can convey sentiment and emotion in text, and can be used as features in sentiment analysis models. For example, you could create a binary feature that indicates whether or not an emoji or emoticon appears in the text, or you could use a sentiment lexicon to assign a sentiment score to each emoji.

Hashtags: Hashtags can be used as features to identify topics or sentiments expressed in social media posts. For example, if a post contains the hashtag #happy, it's likely that the sentiment expressed is positive.

Named Entities: Named entities such as people, organizations, and locations can be used as features to identify the sentiment expressed toward them. For example, if a review mentions a specific restaurant or product, the sentiment expressed toward that entity can be used as a feature.

Sarcasm detection: Sarcasm can be difficult to detect in text, but it can be a valuable feature for sentiment analysis. There are various techniques that can be used to detect sarcasm, such as detecting contrastive elements or looking for negation and hyperbole.

Syntax and Grammar: Syntax and grammar can provide important cues about sentiment in text. For example, negative sentiment may be expressed through the use of negative words or the presence of grammatical constructions such as passive voice or conditional statements.

Sentiment Shift: Sentiment shift can be used to identify changes in sentiment over time or within a text. For example, if a review starts with positive sentiment but then shifts to negative sentiment, this could be used as a feature to predict the overall sentiment of the review.

These are just a few examples of the many features that can be used in sentiment analysis. The most effective features will depend on the specific task and domain, and may require experimentation and fine-tuning to achieve the best performance

VI. Conclusion

The idea behind sentiment analysis project is to analysis the sentiment of the user. Here with simplest expressable model, we have tried to show how python can be a useful platform to develop models for sentiment analysis. Throughout this research, we have tried to focus on general sentiment analysis. There are certain keywords that people want to get the results like on film stars, products, politics or sportsman etc. and to serve the result, performing these general model based on some simple machine learning algorithms would be really beneficial. Simply it can give a much simpler and convenient way to analyze things. Therefore, this article provides an overall knowledge about how sentiment analysis can be easily performed for different type of data to get relevant benefits.

VII. References

[1] Pansy Nandwani et al., “A review on sentiment analysis and emotion detection from text”, Social Network Analysis and Mining (2021) 11:81 https://doi.org/10.1007/s13278-021-00776-6.
[2] R. Venkateswaran et al., “Impact of Social Media Application in Business Organizations”, International Journal of Computer Applications (0975 – 8887) Volume 178 – No. 30, July 2019.
[3] Helen Cripps et al., ”The use of Twitter for innovation in business markets”, Marketing Intelligence & Planning Vol. 38 No. 5, 2020 pp. 587-601 Emerald Publishing Limited 0263-4503 DOI 10.1108/MIP-06-2019-0349
[4] Mitali Desai et al., “Techniques for Sentiment Analysis of Twitter Data: A Comprehensive Survey”, International Conference on Computing, Communication and Automation (ICCCA2016), 2016.
[5] ZulfadzliDrus et al., "Sentiment Analysis in Social Media and Its Application: Systematic Literature Review", Procedia Computer Science 161 (2019) 707–714
[6] Xing Fang et al., " Sentiment analysis using product review data", Journal of Big Data, 5 (2015)
[7] Zhou Gui Zhou, "Research on Sentiment Analysis Model of Short Text Based on Deep Learning", Scientific Programming, 2022
[8] Brian Keith Norambuena et al., Sentiment analysis and opinion mining applied to scientific paper reviews , Intelligent Data Analysis 23 (2019) 191–214

VIII. Biographies

Sayantan Bhattacharya is a final year B.Tech student of the department of Computer Science and Engineering at Netaji Subhash Engineering College in the year 2023.

Mahmadul Hasan is a final year B.Tech student of the department of Computer Science and Engineering at Netaji Subhash Engineering College in the year 2023.

MdRahmat Ali is a final year B.Tech student of the department of Computer Science and Engineering at Netaji Subhash Engineering College in the year 2023.

Fahad Ghayas is a final year B.Tech student of the department of Computer Science and Engineering at Netaji Subhash Engineering College in the year 2023.

Chandra Das received the M.Sc degree in Computer and Information Science from University of Calcutta, Kolkata, India in 2001 and the M.Tech degree in Computer Science and Engg. from the same University in 2003. She received her PhD degree in engineering from Jadavpur University, Kolkata, India in 2011. She is currently an associate professor in the department of Computer Science and Engineering, Netaji Subhash Engineering College, Kolkata, India. Her research interest includes machine learning, bioinformatics, pattern recognition, data mining and natural language processing. She has published over 45 research papers in several international journals and conference proceedings.

Shilpi Bose is currently received the M.Sc degree in Computer and Information Science from University of Calcutta, Kolkata, India in 2002 and the M.Tech degree in Computer Science and Engg. from the same University in 2004. He received his PhD degree in engineering from Jadavpur University, Kolkata, India in 2023. He is currently an assistant professor in the department of Computer Science and Engineering, Netaji Subhash Engineering College, Kolkata, India. His research interest includes machine learning, bioinformatics, pattern recognition, and data mining. He has published over 30 research papers in several international journals and conference proceedings.