SENTIMENT ANALYSIS ON SOCIAL MEDIA

A Project Report submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Kshitij Varshney (201500356)
Vishal Dixit (201500792)
Raman Gupta (201500558)

Group No. : G58

Under the Guidance of

Dr. Zubair Ashraf (Assistant Professor)

Department of Computer Engineering & Applications

Institute of Engineering & Technology

GLA University

Mathura- 281406, INDIA

April, 2024

Department of Computer Engineering and Applications
GLA University, 17 km Stone, NH#2, Mathura-Delhi Road,
P.O. Chaumuhan, Mathura- 281406 (U.P.)

Declaration

I hereby declare that the work which is being presented in the B.Tech. Project **"Sentiment Analysis on Social Media"**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my work carried under the supervision of **Dr. Zubair Ashraf (Assistant Professor).**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign_____________________
Name of Student: KshitijVarshney
University Roll No.: 201500356

Sign _____________________
Name of Student: Vishal Dixit
University Roll No.: 201500792

Sign_____________________
Name of Student: Raman Gupta
University Roll No.: 201500558

Department of Computer Engineering and Applications
GLA University, 17 km Stone, NH#2, Mathura-Delhi Road, P.O. Chaumuhan, Mathura- 281406 (U.P.)

Certificate

This is to certify that the above statements made by the candidate are correct to the best of my/our knowledge and belief.

____

Supervisor

Dr. Zubair Ashraf

Assistant Professor

Dept. of Computer Engg, & App.

____________________	____________________
Project Co-ordinator	**Program Co-ordinator**
(Dr. Mayank Srivastava)	(**Mr. Jitesh Kumar Bhatia**)
Associate Professor	Assistant Professor
Dept. of Computer Engg, & App.	Dept. of Computer Engg, & App.

Date:

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to my committee. It gives us a great sense of pleasure to present the Project Report of the B. Tech Major project undertaken during B. Tech IV Year. This project will be an acknowledgment of the inspiration, drive, and technical assistance that will be contributed to it by me. We owe a special debt of gratitude to **Dr. Zubair Ashraf**, for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal, and for his constant support and guidance to our work.

His sincerity, thoroughness, and perseverance have been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also teach us about the latest industry-oriented technologies. We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation.

Sign ___________________

Name of Student: Kshitij Varshney

University Roll No.: 201500356

Sign __________________

Name of Student: Vishal Dixit

University Roll No.:201500792

Sign _____________________

Name of Student: Raman Gupta

University Roll No.: 201500558

ABSTRACT

Sentiment analysis on Twitter is essential due to its extensive user base and real-time nature. This study explores methodologies and challenges specific to Twitter sentiment analysis, aiming to glean insights into public opinions and attitudes. Twitter's brevity and informal language pose challenges, necessitating advanced natural language processing techniques. Despite this, sentiment analysis offers opportunities for tracking public sentiment toward events, brands, and social issues.

The relevance of sentiment analysis on Twitter extends to various stakeholders, including marketers, journalists, and policymakers. Marketers can leverage sentiment analysis to gauge consumer sentiment towards their products or services, while journalists can monitor public opinion on breaking news stories. Additionally, policymakers can utilize sentiment analysis to assess public response to policy decisions and social issues.

Methodologies commonly employed in sentiment analysis on Twitter encompass lexicon-based approaches and machine-learning algorithms trained on labelled datasets. However, challenges such as context ambiguity, sarcasm, and slang persist in tweets, driving ongoing efforts to enhance the accuracy and robustness of sentiment analysis models.

In conclusion, sentiment analysis on Twitter provides real-time insights into public sentiment. Despite challenges, advancements in natural language processing improve accuracy, facilitating informed decision-making for businesses, media, and policymakers.

CONTENTS

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

With the increase in social awareness; the popularity of social networking such as Twitter is increased. Twitter is one of the important and popular social media where anyone can post tweets about any event. This is an open platform where people may express their views/opinions or emotions freely. Due to lower internet charges, less expensive portable devices, and increased social importance; people have Twitter accounts. Most of them tweet about different events. In the social networking age, people express their opinions and feelings through Twitter. So, Twitter contains a huge amount of data.

Twitter sentiment analysis is one of the recent and challenging research areas. As social media like Twitter contains a huge amount of text sentiment data in the form of tweets it is useful to identify sentiments or opinions of people about specific events. Sentiment analysis or opinion mining is useful for the review of movies, products, customer services, opinions about any event, etc. This helps us to decide whether a specific item or service is good/bad or preferred or not preferred. It is also useful to identify the opinions of people about any event or person and also find the polarity of the text whether positive, negative or neutral. Sentiment analysis is a type of text classification that can classify text into different sentiments.

1.1 OVERVIEW AND MOTIVATION

Sentiment analysis on social media platforms like Twitter is a compelling field of study driven by the unprecedented scale of user-generated content and its potential to offer real-time insights into public sentiment. The sheer volume of data generated on social media daily presents both a challenge and an opportunity for researchers and businesses alike. Understanding the emotions, opinions, and attitudes expressed by users holds immense value across various domains, from marketing and journalism to policymaking and societal research.

One of the primary motivations behind sentiment analysis projects on social media is the need for businesses to comprehend consumer sentiment towards their products or services. In today's highly competitive market landscape, businesses are constantly seeking ways to gain a competitive edge and enhance customer satisfaction. By analyzing sentiment on platforms like Twitter, businesses can gain valuable insights into consumer preferences, identify emerging trends, and tailor their marketing strategies accordingly. This not only helps in improving brand perception but also fosters better engagement with customers, ultimately driving business growth.

Similarly, sentiment analysis projects on social media are motivated by the imperative for journalists to stay attuned to public opinion and interests. With the rapid dissemination of news and information on platforms like Twitter, journalists face the challenge of curating content that resonates with their audience. By harnessing sentiment analysis tools, journalists can gauge public reactions to breaking news stories, identify trending topics, and adjust their reporting to better reflect the interests and concerns of their audience. This not only enhances the relevance and timeliness of their reporting but also strengthens trust and credibility with their readership.

Moreover, sentiment analysis projects on social media are driven by the desire for policymakers to make data-driven decisions that resonate with the sentiments of the public they serve. In an era of heightened political polarization and social activism, understanding public sentiment on key issues is paramount for effective governance. By analyzing sentiments expressed on platforms like Twitter, policymakers can gauge public reactions to policy decisions, anticipate potential backlash, and devise more responsive and inclusive policies. This facilitates a more democratic and participatory decision-making process, ensuring that policies reflect the diverse voices and opinions of the populace.

Overall, sentiment analysis projects on social media are motivated by the profound impact they can have across various domains, from shaping business strategies and journalistic practices to informing policymaking and societal discourse. By harnessing the power of sentiment analysis, stakeholders can gain deeper insights into public sentiment, foster better communication and engagement, and ultimately drive positive outcomes for individuals and society as a whole.

1.2 OBJECTIVE

The primary objective of our project is to develop a machine learning-based system for Sentiment Analysis on Twitter.

i. **Algorithm Development:** The primary objective of our project is to design and develop a machine learning-based system specifically tailored for sentiment analysis on social media, with a focus on Twitter. This involves creating algorithms that can effectively parse and interpret the vast amount of user-generated content on Twitter, categorizing sentiments expressed in tweets as positive, negative, or neutral.

ii. **Model Training and Optimization:** Another key objective is to train and optimize machine learning models using labeled datasets to accurately predict sentiment in Twitter data. This entails exploring various machine learning algorithms, feature engineering techniques, and model architectures to enhance the system's performance in capturing the nuances of sentiment expressed in tweets.

iii. **Real-Time Analysis:** The project aims to implement real-time sentiment analysis capabilities, enabling the system to process and analyze tweets as they are posted on Twitter. This involves developing efficient data processing pipelines and integrating the machine learning models into a scalable and responsive infrastructure that can handle the high volume of Twitter data streams.

iv. **Evaluation and Validation:** Finally, the objective includes rigorously evaluating and validating the performance of the developed system against benchmark datasets and real-world Twitter data. This entails conducting comprehensive experiments to assess the accuracy, precision, recall, and overall effectiveness of the sentiment analysis system, ensuring its reliability and suitability for practical applications in social media analytics.

1.3 ISSUES AND CHALLENGES

In addressing sentiment analysis on social media, several challenges and limitations exist when conducting sentiment analysis on social media, particularly Twitter:

i. **Ambiguity and Contextual Understanding**: Tweets often contain ambiguous language, slang, or cultural references that can be challenging for sentiment analysis algorithms to interpret accurately without context. This ambiguity can lead to misclassification of sentiments, especially in cases of sarcasm or irony.

ii. **Data Sparsity and Noise**: Twitter data is vast and noisy, comprising a mix of relevant tweets, spam, advertisements, and irrelevant content. Noise in the data can introduce biases and affect the performance of sentiment analysis algorithms, requiring robust preprocessing techniques to filter out irrelevant information.

iii. **Short Texts and Informal Language**: The 280-character limit per tweet restricts the amount of information available for sentiment analysis. Furthermore, tweets often contain informal language, abbreviations, and misspellings, making it challenging for algorithms to accurately capture sentiments expressed in short and sometimes cryptic messages.

iv. **Subjectivity and Variability**: Sentiment analysis is inherently subjective, as the perception of sentiment can vary among individuals and across different contexts. Moreover, sentiments expressed on Twitter can be influenced by factors such as cultural background, demographics, and personal beliefs, adding complexity to the analysis process.

v. **Class Imbalance and Minority Sentiments**: Sentiment analysis datasets on Twitter often exhibit class imbalance, where one sentiment class is predominant while others are underrepresented. This imbalance can skew the performance of sentiment analysis models, particularly in accurately detecting minority sentiments, requiring techniques to address class imbalance such as data augmentation or resampling.

1.4 CONTRIBUTION

Our project addresses several challenges in sentiment analysis on social media by leveraging machine learning techniques and innovative approaches. The following contributions highlight how our project aimed to address challenges in sentiment analysis:

i. **Enhanced Accuracy through Machine Learning**: Your project aims to improve the accuracy of sentiment analysis by leveraging advanced machine learning techniques. By training models on large datasets of labelled Twitter data, your project seeks to develop more robust algorithms capable of accurately identifying and classifying sentiments expressed in tweets.

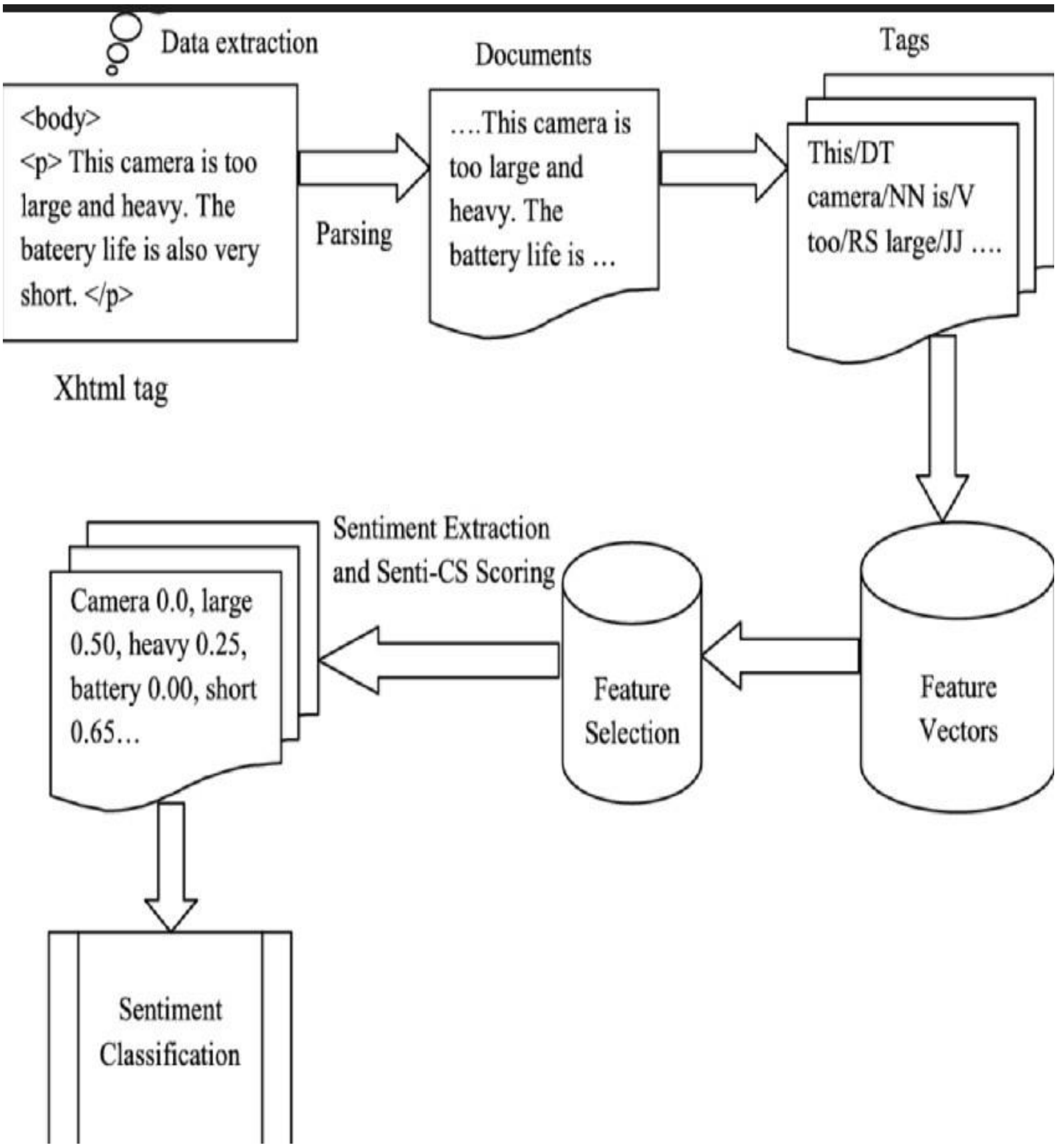
ii. **Handling Short Texts and Informal Language**: One of the challenges in sentiment analysis on Twitter is dealing with short texts and informal language. Your project aims to address this challenge by developing innovative approaches to extract sentiment from succinct and often informal tweets, considering factors like slang, abbreviations, and misspellings.

iii. **Contextual Understanding and Ambiguity**: Sentiment analysis often struggles with understanding the context and ambiguity present in tweets, such as sarcasm or irony. Your project aims to overcome this challenge by exploring techniques to incorporate contextual information and disambiguate sentiments, enabling more accurate analysis of nuanced language.

iv. **Real-Time Analysis and Scalability**: Your project seeks to enable real-time sentiment analysis on Twitter data, allowing for timely insights into public sentiment. By developing scalable and efficient algorithms and infrastructure, your project aims to process large volumes of tweets in real-time, providing valuable insights for businesses, journalists, and policymakers.

v. **Open-Source Contributions and Knowledge Sharing**: Lastly, your project aims to contribute to the advancement of sentiment analysis by sharing code implementations, datasets, and research findings with the wider community. By fostering collaboration and knowledge sharing, your project seeks to accelerate progress in sentiment analysis and encourage the adoption of innovative approaches across the field.

1.5 BASIC ARCHITECTURE OF THE PROJECT:



1.1 Basic Architecture of the project

1.6 ORGANIZATION OF THE PROJECT REPORT

- Introduction
- Literature Review
- Proposed Work
- Implementation and Result Analysis
- Conclusion
- References

CHAPTER 2

LITERATURE REVIEW

2.1 LIMITATIONS OF PRIOR ART

Prior art in sentiment analysis on social media platforms like Twitter exhibits several noteworthy limitations. Firstly, traditional approaches often struggle with the brevity and informality of tweets, making it challenging to accurately interpret sentiments expressed in short and sometimes cryptic messages. This issue is compounded by the prevalence of slang, abbreviations, and misspellings in Twitter data, which can lead to misinterpretations of sentiment. Moreover, traditional sentiment analysis models frequently fail to capture the nuanced linguistic features present in tweets, such as sarcasm, irony, or ambiguity, which can significantly impact the accuracy of sentiment classification. Another limitation lies in the domain-specificity of many sentiment analysis systems, as they may not generalize well to diverse topics or datasets outside their training domain, thus limiting their applicability. Furthermore, scalability is a concern, with many prior approaches unable to efficiently process the massive volumes of Twitter data in real time, hindering their effectiveness in dynamic online environments. Additionally, the predominance of text-based analysis overlooks the rich multimodal content shared on Twitter, including images, videos, and emoji's, which can convey sentiment in ways not captured by traditional text-based approaches. These limitations underscore the pressing need for innovative solutions that can effectively address the unique challenges of sentiment analysis on Twitter, enhancing the accuracy, scalability, and comprehensiveness of sentiment analysis on this platform.

2.2 RELATED WORK

The bag-of-words model is one of the most widely used feature model for almost all text classification tasks due to its simplicity coupled with good performance. The model represents the text to be classified as a bag or collection of individual words with no link or dependence of one word with the other, i.e. it completely disregards grammar and order of words within the text. This model is also very popular in sentiment analysis and has been used by various researchers. The simplest way to incorporate this model in our classifier is by using unigrams as features. Generally speaking n-grams is a contiguous sequence of "n" words in our text, which is completely independent of any other words or grams in the text. So unigrams is just a collection of individual words in the text to be classified, and we assume that the probability of occurrence of one word will not be affected by the presence or absence of any other word in the text. This is a very simplifying assumption but it has been shown to provide rather good performance. One simple way to use unigrams as features is to assign them with a certain prior polarity, and take the average of the overall polarity of the text, where the overall polarity of the text could simply be calculated by summing the prior polarities of individual unigrams. Prior polarity of the word would be positive if the word is generally used as an indication of positivity, for example the word "sweet"; while it would be negative if the word is generally associated with negative connotations, for example "evil". There can also be degrees of polarity in the model, which means how much indicative is that word for that particular class. A word like "awesome" would probably have strong subjective polarity along with positivity, while the word "decent" would although have positive prior polarity but probably with weak subjectivity.

There are three ways of using prior polarity of words as features. The simpler un-supervised approach is to use publicly available online lexicons/dictionaries which map a word to its prior polarity. The Multi-Perspective-Question-Answering (MPQA) is an online resource with such a subjectivity lexicon which maps a total of 4,850 words according to whether they are "positive" or "negative" and whether they have "strong" or "weak" subjectivity. The SentiWordNet 3.0 is another such resource which gives probability of each word belonging to positive, negative and neutral classes. The second approach is to construct a custom prior polarity dictionary from our training data according to the occurrence of each word in each particular class. For example if a certain word is occurring more often in the positive labelled phrases in our training dataset (as compared to other classes) then we can calculate the probability of that word belonging to positive class to be higher than the probability of occurring in any other class. This approach has been shown to give better performance, since the prior polarity of words is more suited and fitted to a particular type of text and is not very general like in the former approach. However, the latter is a supervised approach because the training data has to be labelled in the appropriate classes before it is possible to calculate the relative occurrence of a word in each of the class. Kouloumpis et al. noted a decrease in performance by using the lexicon word features along with custom n-gram word features constructed from the training data, as opposed to when the n-grams were used alone.

The third approach is a middle ground between the above two approaches. In this approach we construct our own polarity lexicon but not necessarily from our training data, so we don't need to have labelled training data. One way of doing this as proposed by Turney et al. is to calculate the prior semantic orientation (polarity) of a word or phrase by calculating it's mutual information with the word "excellent" and subtracting the result with the mutual information of that word or phrase with the word "poor". They used the number of result hit counts from online search engines of a relevant query to compute the mutual information. The final formula they used is as follows:

$$\text{Polarity (phrase)} = \log \frac{hit(phrase\ NEAR\ \text{"excellent"}).\, hit(\text{"poor"})}{hit(phrase\ NEAR\ \text{"poor"}).\, hits(\text{"}ecellent\text{"})}$$

Where *hits (phrase NEAR "excellent")* means the number documents returned by the search engine in which the phrase (whose polarity is to be calculated) and word "excellent" are co-occurring. While *hits ("excellent")* means the number of documents retuned which contain the word "excellent". Prabowo et al. have gone ahead with this idea and used a seed of 120 positive words and 120 negative to perform the internet searches. So the overall semantic orientation of the word under consideration can be found by calculating the closeness of that word with each one of the seed words and taking and average of it. Another graphical way of calculating polarity of adjectives has been discussed by Hatzivassiloglou et al,. The process involves first identifying all conjunctions of adjectives from the corpus and using a supervised algorithm to mark every pair of adjectives as belonging to the same semantic orientation or different. A graph is constructed in which the nodes are the adjectives and links indicate same or different semantic orientation. Finally a clustering algorithm is applied which divides the graph into two subsets such that nodes within a subset mainly contain links of same orientation and links between the two subsets mainly contain links of different orientation. One of the subsets would contain positive adjectives and the other would contain negative.

Many of the researchers in this field have used already constructed publicly available lexicons of sentiment bearing words while many others have also explored building their own prior polarity lexicons.

The basic problem with the approach of prior polarity approach has been identified by Wilson et al. who distinguish between prior polarity and contextual polarity. They say that the prior polarity of a word may in fact be different from the way the word has been used in the particular context. The paper presented the following phrase as an example: Philip Clapp, president of the National Environment Trust, sums up well the general thrust of the reaction of environmental movements: “There is no reason at all to believe that the polluters are suddenly going to become reasonable.”

In this example all of the four underlined words “trust”, “well”, “reason” and “reasonable” have positive polarities when observed without context to the phrase, but here they are not being used to express a positive sentiment. This concludes that even though generally speaking a word like “trust” may be used in positive sentences, but this doesn’t rule out the chances of it appearing in non-positive sentences as well. Henceforth prior polarities of individual words (whether the words generally carry positive or negative connotations) may alone not enough for the problem. The paper explores some other features which include grammar and syntactical relationships between words to make their classifier better at judging the contextual polarity of the phrase.

The task of twitter sentiment analysis can be most closely related to phrase- level sentiment analysis. A seminal paper on phrase level sentiment analysis was presented in 2005 by Wilson et al. which identified a new approach to the problem by first classifying phrases according to subjectivity (polar) and objectivity (neutral) and then further classifying the subjective-classified phrases as either positive or negative. The paper noticed that many of the objective phrases used prior sentiment bearing words in them, which led to poor classification of especially objective phrases. It claims that if we use a simple classifier which assumes that the contextual polarity of the word is merely equal to its prior polarity gives a result of about 48%. The novel classification process proposed by this paper along with the list of ingenious features which include information about contextual polarity resulted in significant improvement in performance (in terms of accuracy) of the classification process. The results from this paper are presented in the table below:

Features	Accuracy	Subjective F.	Objective F.
Word tokens	73.6	55.7	81.2
Words + prior polarity	74.2	60.6	80.7
28 features	75.9	63.6	82.1

Table 1: Results for Objective / Subjective Classification

Features	Accuracy	Positive F.	Negative F.	Both F.	Objective F.
Word tokens	61.7	61.2	73.1	14.6	37.7
Word + prior	63.0	61.6	75.5	14.6	40.7
10 features	65.7	65.1	77.2	16.1	46.2

Table 2: Results for Polarity Classification

One way of alleviating the condition of independence and including partial context in our word models is to use bigrams and trigrams as well besides unigrams. Bigrams are collection of two contiguous words in a text, and similarly trigrams are collection of three contiguous words. So we could calculate the prior polarity of the bigram / trigram - or the prior probability of that bigram / trigram belonging to a certain class – instead of prior polarity of individual words. Many researchers have experimented with them with the general conclusion that if we have to use one of them alone unigrams perform the best, while unigrams along with bigrams may give better results with certain classifiers. However trigrams usually result in poor performance as reported by Pak et al.. The reduction in performance by using trigrams is because there is a compromise between capturing more intricate patterns and word coverage as one goes to higher-numbered grams. Besides from this some researchers have tried to incorporate negation into the unigram word models. Pang et al. and Pakl et al. used a model in which the prior polarity of the word was reserved if there was a negation (like “not”, “no”, “don’t”, etc.) next to that word. In this way some contextual information is included in the word models.

Grammatical features (like “Parts of Speech Tagging” or POS tagging) are also commonly used in this domain. The concept is to tag each word of the tweet in terms of what part of speech it belongs to: noun, pronoun, verb, adjective, adverb, interjections, intensifiers etc. The concept is to detect patterns based on these POS and use them in the classification process. For example it has been reported that objective tweets contain more common nouns and third-person verbs than subjective tweets, so if a tweet to be classified has a proportionally large usage of common nouns and verbs in third person, that tweet would have a greater probability of being objective (according to this particular feature). Similarly subjective tweets contain more adverbs, adjectives and interjections. These relationships are demonstrated in the figures below:

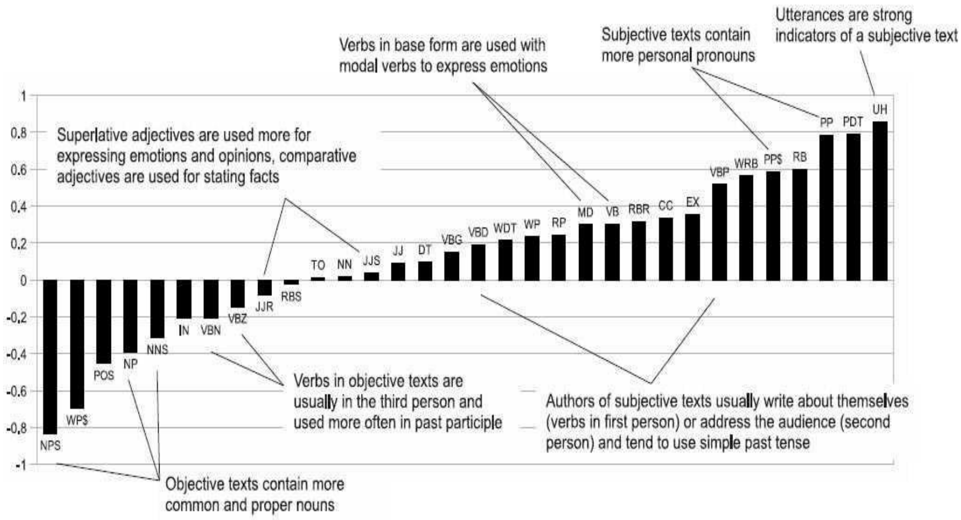


Figure 1: Using POS Tagging as features for objectivity/subjectivity classification

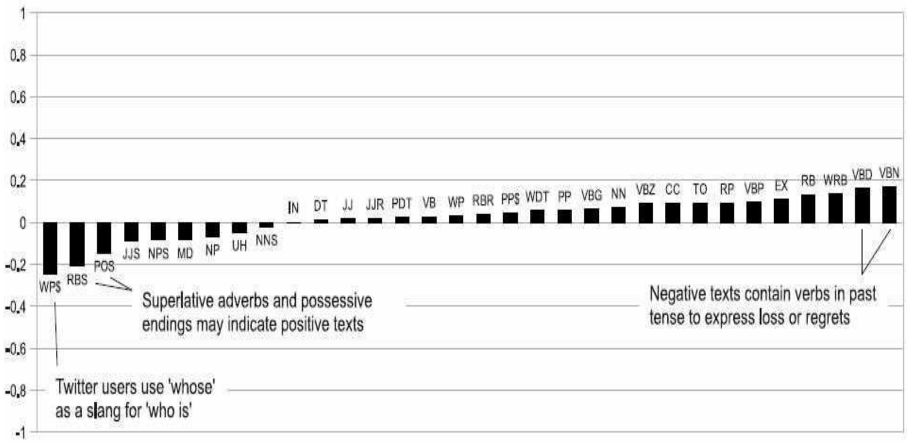


Figure 2: Using POS Tagging as features in positive/negative classification

However, there is still conflict whether Parts-of-Speech are a useful feature for sentiment classification or not. Some researchers argue in favor of good POS features while others not recommending them.

Besides from these much work has been done in exploring a class of features pertinent only to micro-blogging domain. The presence of URLs and several capitalized words/alphabets in a tweet have been explored by Koulompis et al. and Barbosa et al.. Koulmpis also reports positive results for using emoticons and internet slang words as features. Brody et al. does study on word lengthening as a sign of subjectivity in a tweet. The paper reports positive results for their study that the more number of cases a word has of lengthening, the more chance there of that word being a strong indication of subjectivity.

The most commonly used classification techniques are the Naive Bayes Classifier and State Vector Machines. Some researchers like Barbosa et al. publish better results for SVMs while others like Pak et al. support Naive Bayes and also report good results for Maximum Entropy classifier.

It has been observed that having a larger training sample pays off to a certain degree, after which the accuracy of the classifier stays almost constant even if we keep adding more labelled tweets in the training data. Barbosa et al. used tweets labelled by internet resources, instead of labelling them by hand, for training the classifier. Although there is loss of accuracy of the labelled samples in doing so (which is modelled as increase in noise) but it has been observed that if the accuracy of training labels is greater than 50%, the more the labels, the higher the accuracy of the resulting classifier. So in this way if there are an extremely large number of tweets, the fact that our labels are noisy and inaccurate can be compensated for. On the other hand Pak et al. and Go et al. use presence of positive or negative emoticons to assign labels to the tweets. Like in the above case they used large number of tweets to reduce effect of noise in their training data.

Some of the earliest work in this field classified text only as positive or negative, assuming that all the data provided is subjective. While this is a good assumption for something like movie reviews but when analyzing tweets and blogs there is a lot of objective text we have to consider, so incorporating neutral class into the classification process is now becoming a norm. Some of the work which has included neutral class into their classification process includes.

There has also been very recent research of classifying tweets according to the mood expressed in them, which goes one step further. Bollen et al. explores this area and develops a technique to classify tweets into six distinct moods: tension, depression, anger, vigour, fatigue and confusion. They use an extended version of Profile of Mood States (POMS): a widely accepted psychometric instrument. They generate a word dictionary and assign them weights corresponding to each of the six mood states, and then they represented each tweet as a vector corresponding to these six dimensions. However not much detail has been provided into how they built their customized lexicon and what technique did they use for classification.

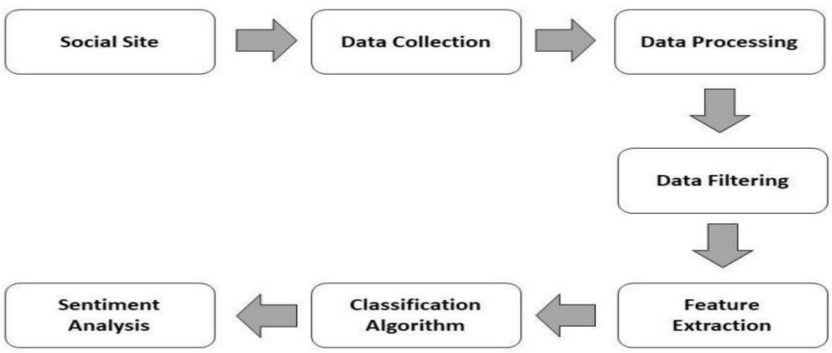
CHAPTER 3

SYSTEM DESIGN

A sentiment analysis system for social media entails a multi-faceted approach beginning with data collection through APIs or web scraping, followed by preprocessing steps such as tokenization, normalization, and feature extraction including Bag-of-Words, TF-IDF, or word embeddings. Model selection involves choosing between machine learning or deep learning techniques like SVM, Naive Bayes, RNNs, CNNs, or Transformer-based models, which are then trained on labeled data and evaluated for performance metrics. During inference, the trained model predicts sentiment on unseen data, with deployment on scalable infrastructure and integration via APIs. Continuous feedback loops ensure model refinement and ethical considerations address bias, fairness, transparency, and privacy, while scalability and maintenance ensure robustness and efficiency over time.

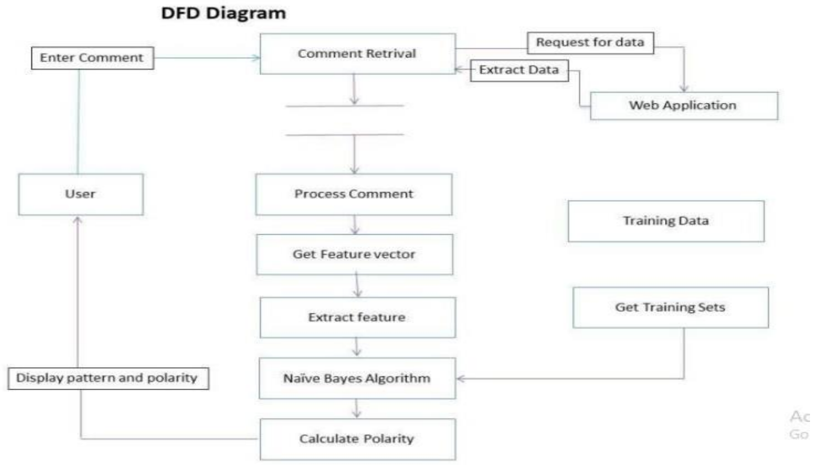
1. SYSTEM ARCHITECTURE:

A minimum of one author is required for all conference articles. Author names should be listed startingfrom left to right and then moving down to the next line. This is the author sequence that will be used in futurecitations and by indexing services. Names should not be listed in columns nor grouped by affiliation. Please keepyour affiliations as succinct as possible (for example, do not differentiate among departments of the same organization)



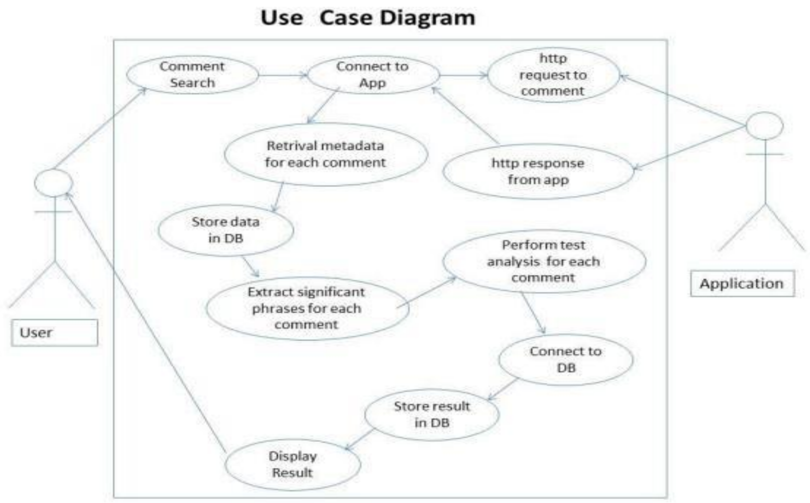
2. Data Flow Diagram:

In Data Flow Diagram, we Show that flow of data in our system in DFD0 we show that base DFD in whichrectangle present input as well as output and circle show our system, In DFD1 we show actual input and actualoutput of system input of our system is text or image and output is rumor detected like wise in DFD 2 we present operation of user as well as admin.



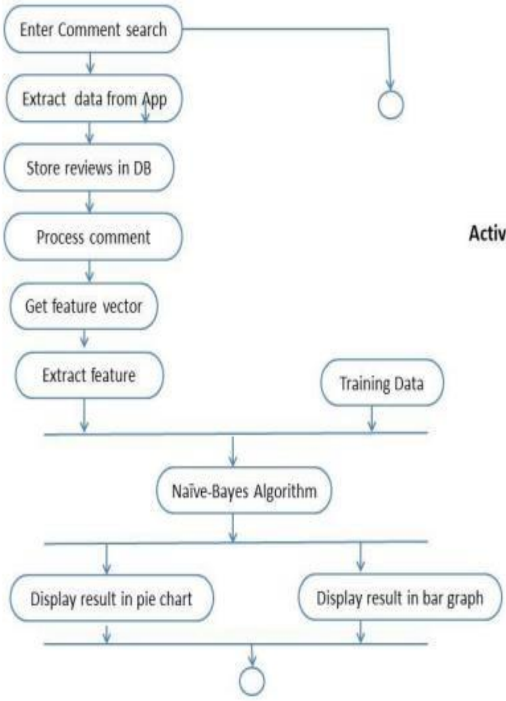
3. Use Case Diagrams:

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagramcan identify the different types of users of a system and the different use cases and will often be accompaniedby other types of diagrams as well. The use cases are represented by either circles or ellipses.



4. Activity Diagram:

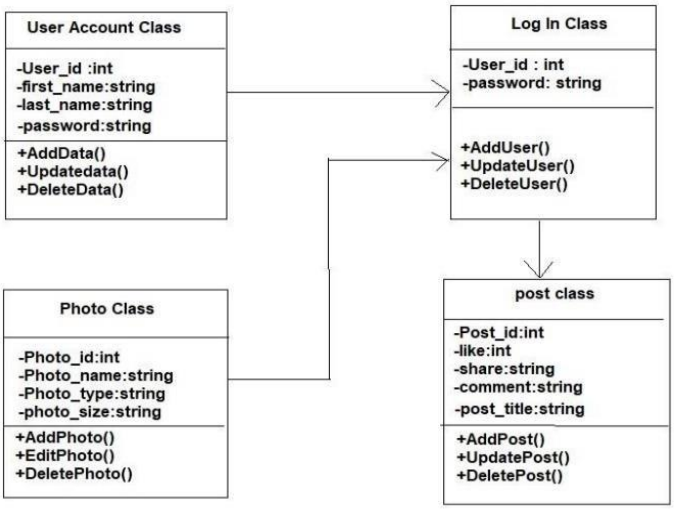
Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e workflows), as well as the data flowsintersecting with the related activities. Although activity diagrams primarily show the overall flow of controlthey can also include elements showing the flow of data between activities through one or more data stores.



Activity Diagram

5. Class Diagram:

Social Networking Site Class Diagram describes the structure of a Social Networking Site classes, theirattributes, operations (or methods), and the relationships among objects. The main classes of the Social Networking Site are Users, Comment, Posts, Shares, Photos are intended to model both computational and organizational processes (i.e workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control they can also include



6. UML Diagram:

Unified Modeling Language is a standard language for writing software blueprints. The UML may be used tovisualize, specify, construct and document the artifacts of a software-intensive system is process independent,although optimally it should be used in the process that is use case driven, architecture-centric, iterative, and increment. The Number of UML diagrams is available. Use case Diagram. Activity Diagram. Sequence Diagram.

CHAPTER 4

PROPOSED WORK

This chapter outlines the proposed approach for sentiment analysis on social media using machine learning techniques. The methodology involves several key steps, including data preprocessing, model selection, optimization, feature selection, and model refinement.The proposed approach aims to improve accuracy and efficiency by leveraging machine learning algorithms and optimization techniques.

1. **Data Acquisition:** To gather data for our sentiment analysis, we utilized the Python library "twee stream" to access Twitter's streaming API. This API offers two modes for accessing tweets: Sample Stream and Filter Stream. Sample Stream provides a random sample of all streaming tweets, while Filter Stream delivers tweets that match specific criteria such as keywords, user IDs, or geo-locations. For our study, we opted to use the Sample Stream mode to ensure a broad representation of tweets. To enhance the generality of our data, we collected tweets at different points in time instead of all at once. Acquiring all tweets simultaneously might have led to a biased dataset, as a large portion would likely focus on trending topics, skewing the overall sentiment. Sampling our data at different times aimed to mitigate this issue. Therefore, we collected data on different dates. Each tweet acquired through this method contains various key-value pairs in a Python dictionary format, providing information such as whether the tweet was favorited, user ID, user screen name, the original text of the tweet, presence of hashtags, retweet status, user's account language, geo-tag location of the tweet, and the tweet's creation date and time. Due to the abundance of information, we filtered out irrelevant data and focused only on the tweet text content. For our application, we saved the text content of tweets where the user's account language was specified as English. The original tweet content is stored under the dictionary key "text," and the user's account language is under "lang." To reduce the cost of human labeling, we further refined our tweet selection criteria to maximize variation while maintaining generality. We

filtered out retweets, very short tweets (less than 20 characters), non-English tweets (by comparing tweet content with a list of 2,000 common English words), and similar tweets (by comparing content similarity between tweets). After these filters, approximately 30% of the tweets remained for human labeling.

2. **Human Labelling:** We created three copies of the tweets to be labeled for the human labeling process, allowing four individual sources to label them. This approach was chosen to minimize noise and inaccuracies by averaging the opinions of multiple people. While more copies would have been ideal, we opted for three to balance the cost of labeling.

The tweets were categorized into four classes based on the sentiments expressed or observed: positive, negative, neutral/objective, and ambiguous. Guidelines were provided to the labelers for each category:

- **Positive**: Entire tweet has a positive/happy/excited/joyful attitude, or if something is mentioned with positive connotations. If more than one sentiment is expressed, the positive sentiment is more dominant.
- **Negative**: Entire tweet has a negative/sad/displeased attitude, or if something is mentioned with negative connotations. If more than one sentiment is expressed, the negative sentiment is more dominant.
- **Neutral/Objective**: The creator expresses no personal sentiment/opinion in the tweet, and merely transmits information. Advertisements of different products are labelled under this category.
- **Ambiguous**: More than one sentiment is expressed in the tweet with no one particular sentiment standing out. If it's obvious that some personal opinion is being expressed but due to lack of context, it is difficult/impossible to accurately decipher the sentiment. Also, if the context of the tweet is not apparent from the information available.

Labellers were instructed to remain unbiased, make no assumptions, and judge the tweet only based on the information provided in the tweet itself.

After labelling, the opinions of three labellers were combined to get an average opinion using a majority vote. If all three labels were different, the tweet was labelled as "unable to reach a majority vote."

The final statistics for each class after majority voting were:

- Positive: 2543 tweets
- Negative: 1877 tweets
- Neutral: 4543 tweets
- Ambiguous: 451 tweets
- Unable to reach majority vote: 390 tweets
- Unlabelled non-English tweets: 369 tweets

Considering only tweets with a positive, negative, or neutral majority vote, we were left with 8963 tweets for our training set. Out of these, 4543 were objective tweets, and 4420 were subjective tweets (sum of positive and negative tweets).

We also calculated human-human agreement for our labelling task, resulting in the following agreement percentages:

These results indicate that sentiment classification is inherently a difficult task, even for human beings, as shown by the relatively moderate agreement percentages.

	Human 1: Human 2	Human 2: Human 3	Human 1: Human 3
Strict	58.9%	59.9%	62.5%
Lenient	65.1%	67.1%	73.0%

Table 3: Human-human Agreement in Tweet Labelling

3. **Feature Extraction:** Now that we have arrived at our training set we need to extract useful features from it which can be used in the process of classification. But first, we will discuss some text formatting techniques that will aid us in feature extraction:

    - **Tokenization:** It is the process of breaking a stream of text into words, symbols, and other meaningful elements called "tokens". Tokens can be separated by whitespace characters and/or punctuation characters. It is done so that we can analyse tokens as individual components that make up a tweet.

- **URLs and User References Removal:** Eliminating URLs and user references (e.g., @username) if we are only interested in analysing the text of the tweet.
- **Punctuation Marks and Digits/Numerals Removal:** Removing punctuation marks and digits/numerals, for example, to compare the tweet to a list of English words.
- **Lowercase Conversion:** Normalizing tweets by converting them to lowercase, which makes their comparison with an English dictionary easier.
- **Stemming:** It is the process of reducing a derived word to its root or stem. For example, a stemmer would reduce the phrases "stemmer", "stemmed", "stemming" to the root word "stem". The advantage of stemming is that it simplifies comparison between words, as we do not need to deal with complex grammatical transformations of the word. In our case, we employed the "Porter stemming" algorithm on both the tweets and the dictionary whenever there was a need for comparison.
- **Stop-words Removal:** Stop words are a class of extremely common words that hold no additional information when used in a text and are thus considered useless. Examples include "a", "an", "the", "he", "she", "by", "on", etc. It is sometimes convenient to remove these words because they hold no additional information since they are used almost equally in all classes of text. For example, when computing prior sentiment polarity of words in a tweet according to their frequency of occurrence in different classes and using this polarity to calculate the average sentiment of the tweet over the set of words used in that tweet.
- **Parts-of-Speech Tagging:** POS-Tagging is the process of assigning a tag to each word in the sentence to indicate which grammatical part of speech that word belongs to, i.e., noun, verb, adjective, adverb, coordinating conjunction, etc.

Now we considered various aspects of tweets to differentiate between different classes: objectivity/subjectivity and positivity/negativity.

For objectivity/subjectivity classification, we explored features such as the presence of punctuation marks like exclamation marks and question marks, URLs, and emoticons.

We also utilized unigram word models calculated using Naive Bayes, the prior polarity of words from the MPQA lexicon, and counts of digits, capitalized words, and punctuation marks in tweets. Additionally, we looked at the ratio of non-dictionary words to total words, tweet length, and counts of different types of verbs, adjectives, adverbs, and pronouns in various forms.

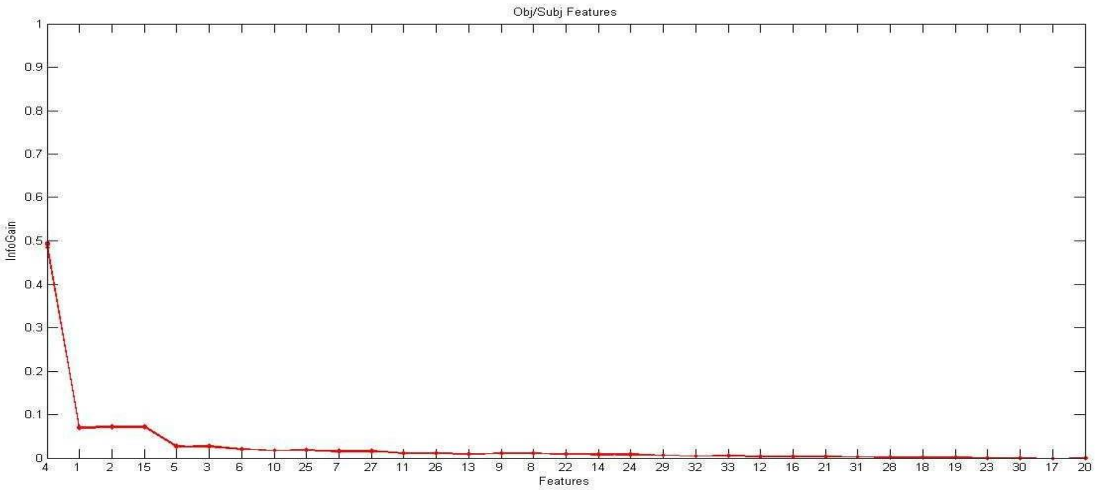


Fig 3: Information Gain of Objectivity / Subjectivity Features

This graph is basically the super-imposition of 10 different graphs, each one arrived through one fold out of the 10-fold cross validation we performed. Since we see that all the graphs are nicely overlapping so the results each fold are almost the same which shows us that the features we select will perform best in all the scenarios. We selected the best 5 features from this graph which are as follows:

1. Unigram word models (for prior probabilities of words belonging to objective / subjective classes)
2. Presence of URL in tweet
3. Presence of emoticons in tweet
4. Number of personal pronouns in tweet
5. Number of exclamation marks in tweet

For positivity/negativity classification, we considered an overall emoticon score, the overall score from the MPQA lexicon, and counts of total, positive, and negative emotions. We also used unigram word models calculated using Naive Bayes, counts of positive and negative words from the MPQA lexicon, and counts of different types of verbs, nouns, pronouns, and adverbs.

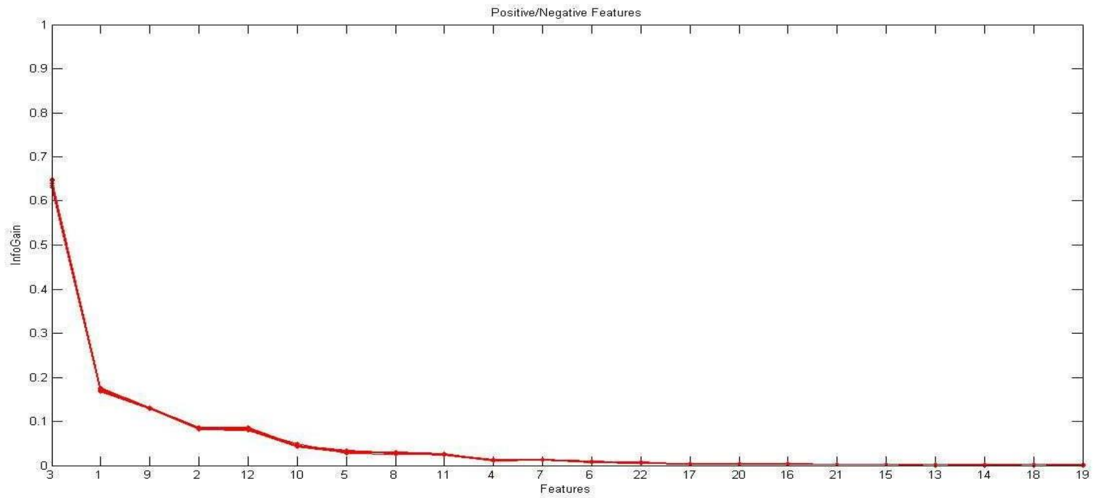


Fig 4: Information Gain of Positive / Negative (Polarity) Features

This graph is basically the super-imposition of 10 different graphs, each one arrived through one fold out of the 10-fold cross validation we performed. Since we see that all the graphs are nicely overlapping so the results each fold are almost the same which shows us that the features we select will perform best in all the scenarios. We selected the best 5 features out of which 2 were redundant features and we were left with only 3 features for our positive / negative classification which are as follows:

1. Unigram word models (for prior probabilities of words belonging to positive or negative classes)
2. Number of positive emoticons in tweet
3. Number of negative emoticons in tweet

These features were selected to capture different aspects of tweets that could help in distinguishing between objective and subjective tweets, as well as positive and negative sentiment in tweets.

4. Classification: Pattern classification involves grouping data into different classes based on common patterns found within each class, which differ from patterns in other classes. Our project aims to design a classifier that accurately categorizes tweets into four sentiment classes: positive, negative, neutral, and ambiguous.

In sentiment analysis, there are two main approaches: contextual sentiment analysis and general sentiment analysis. Contextual sentiment analysis focuses on classifying specific parts of a text based on context, while general sentiment analysis looks at the overall sentiment of the entire text. Our project specifically focuses on general sentiment analysis of tweets as a whole.

The typical classification approach in this field involves a two-step process. First, Objectivity Classification is performed to determine if a tweet is objective or subjective. Then, Polarity Classification is done (only on subjective tweets) to determine if the sentiment is positive, negative, or both. This two-step approach, as proposed by Wilson et al., has shown improved accuracy compared to a single-step approach.

We propose a slightly different approach:
In the first step, each tweet is subjected to two classifiers: the objectivity classifier and the polarity classifier. The objectivity classifier categorizes tweets as objective or subjective, while the polarity classifier determines if the sentiment is positive or negative. Using the selected features, we apply the Naive Bayes algorithm to assign two numbers (from 0 to 1) to each tweet, representing the probabilities of it belonging to the objective and positive classes. The subjective and negative probabilities can be derived by subtracting these values from 1.

In the second step, these two probabilities are treated as separate features for another classification, where the feature size is just 2. We utilize machine learning algorithms such as K-Means Clustering, Support Vector Machine, Logistic Regression, K Nearest Neighbors, Naive Bayes, and Rule-Based Classifiers using WEKA to achieve the best classification results.

- **K-Means Clustering**: While typically used for clustering tasks, K-Means can also be applied to classification problems by assigning each data point to the cluster it is closest to. It works well when the number of clusters is known and the data is well-separated.
- **Support Vector Machine (SVM)**: SVM is a powerful algorithm for binary classification. It works by finding the hyperplane that best separates the classes in the feature space. SVM is effective in high-dimensional spaces and when the number of features exceeds the number of samples.
- **Logistic Regression**: Despite its name, logistic regression is a linear model for binary classification. It estimates probabilities using a logistic function and makes predictions based on these probabilities. It's simple, fast, and interpretable.

- **K Nearest Neighbours (KNN)**: KNN is a simple, instance-based learning algorithm. It classifies new data points based on the majority class of their k nearest neighbours. KNN is non-parametric and lazy, meaning it doesn't make strong assumptions about the form of the underlying data distribution.
- **Naive Bayes**: Naive Bayes is a probabilistic classifier based on Bayes' theorem and the assumption of independence between features. Despite its simplicity, it often performs well in text classification tasks, like sentiment analysis.
- **Rule-Based Classifiers**: Rule-based classifiers use a set of if-then rules to classify data. These rules are typically derived from the data using techniques like decision tree induction. Rule-based classifiers are often easy to interpret but may struggle with complex relationships in the data.

In the context of sentiment analysis, these algorithms can be used to classify tweets into positive, negative, neutral, and ambiguous categories based on the features extracted from the text. The choice of algorithm depends on the complexity of the sentiment analysis task and the nature of the dataset.

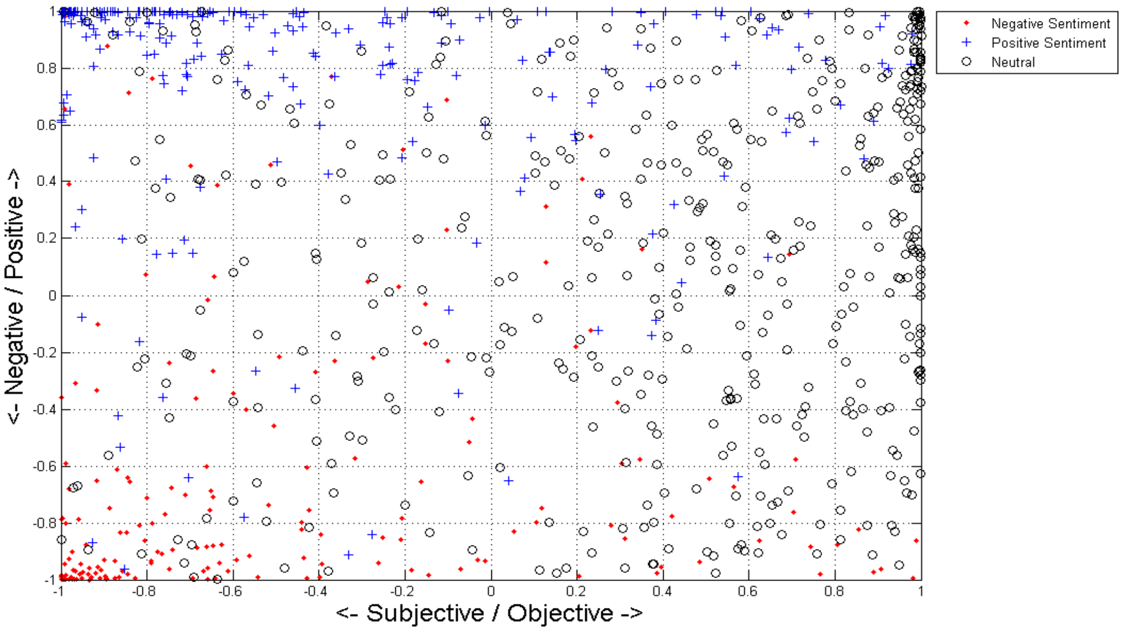


Fig 5: 2d Scatter-Plot

In this figure the labels are the actual ground truth and the distribution shows how the classified data points are actually scattered throughout the space. As we go right the tweet starts becoming increasingly objective and as we go up the tweet starts becoming more positive.

CHAPTER 5

IMPLEMENTATION AND RESULT ANALYSIS

This section provides a comprehensive overview of the implementation steps for sentiment analysis Om social media, along with an in-depth analysis of the obtained results.

Data Collection

For data collection, acquire a dataset containing tweets along with sentiment labels (positive, negative, neutral). This dataset is crucial for training and evaluating the sentiment analysis model. It should encompass a diverse range of tweets to ensure the model's effectiveness across various types of text.

Data Preprocessing

Preprocessing involves several steps to clean the tweets for analysis. Tokenization breaks the text into individual words or tokens. URLs and user references are removed. Punctuation and digits are also eliminated. Converting text to lowercase ensures consistency. Stemming reduces words to their root form. Stop words, such as "a" and "the," are removed as they carry little meaning. This process standardizes the text, making it easier for the model to analyze and extract features.

Objectivity Classification

In the Objectivity Classification phase, a Naive Bayes classifier is employed to categorize tweets as either objective or subjective. This classification is based on various extracted features, including the presence of emoticons, URLs, and the frequency of punctuation marks. By analyzing these features, the classifier determines the likelihood of a tweet being subjective, which indicates personal opinion or sentiment, as opposed to objective, which presents factual information. This step is crucial in distinguishing between tweets that express opinions or sentiments and those that convey factual information, laying the foundation for further sentiment analysis.

Polarity Classification

In the Polarity Classification phase, subjective tweets identified in the Objectivity Classification step are further analyzed to determine their sentiment polarity—whether they are positive, negative, or neutral. This process involves using another Naive Bayes classifier, which is trained on features extracted from the subjective tweets. These features may include the presence and frequency of positive and negative emotions, the occurrence of words with known sentiment from lexicons, and the overall structure and composition of the tweet.

The Naive Bayes classifier calculates the probabilities of a tweet being positive, negative, or neutral based on these features. For example, if a tweet contains many positive words and emoticons, the classifier might assign a higher probability to it being positive. Conversely, if a tweet contains negative words or emoticons, it might be classified as negative. Tweets with a balance of positive and negative features might be categorized as neutral.

By applying this classification process to subjective tweets, the model can effectively determine the sentiment expressed in each tweet, providing valuable insights into the overall sentiment trends within the dataset.

Combining Results

First, we present the results of our objective/subjective and positive/negative classifications, which constitute the initial step of our classification approach. We only utilize the shortlisted features for both classifications, resulting in 5 features for objective/subjective and 3 features for positive/negative classification. We employ the Naïve Bayes classification algorithm for both, as it aligns with our actual classification approach's first step. Additionally, all reported figures are the outcomes of 10-fold cross-validation, where we average the values obtained from each fold.

Classes	True Positive	False Positive	Recall	Precision	F-measure
Objective	0.73	0.26	0.74	0.73	0.73
Subjective	0.74	0.27	0.725	0.73	0.73
Average	0.73	0.27	0.73	0.73	0.73

Table 4: Results from Objective / Subjective Classification

Classes	True Positive	False Positive	Recall	Precision	F-measure
Positive	0.84	0.19	0.86	0.84	0.85
Negative	0.81	0.16	0.79	0.81	0.80
Average	0.83	0.18	0.83	0.83	0.83

Table 5: Results from Polarity Classification (Positive / Negative)

Furthermore, it's important to note that when reporting the results of polarity classification, which distinguishes between positive and negative classes, we impose a condition that only tweets labelled as subjective are used for these calculations. However, in our final classification approach, we remove this condition, and both objectivity and polarity classifications are applied to all tweets, regardless of their labelled objectivity or subjectivity.

Comparing our results to those provided by Wilson et al. [16], we note that although the accuracy of the neutral class decreases from 82.1% to 73% when using our classification instead of theirs, we report significantly higher results for all other classes. While Wilson et al.'s results are not derived from Twitter data, they pertain to phrase-level sentiment analysis, which is closely related in concept to Twitter sentiment analysis.
Next, we compare our results with those presented by Go et al. [2], as shown in the table below:

Features	Naive Bayes	Max Entropy	SVM
Unigram	81.3%	80.5%	82.2%
Bigram	81.6%	79.1%	78.8%
Unigram + Bigram	82.7%	83.0%	81.6%
Unigram + POS	79.9%	79.9%	81.9%

Table 6: Positive / Negative Classification Results

When comparing these results to ours, we find them to be quite similar. However, we achieve comparable results using just 10 features and about 9,000 training data points. In contrast, they used about 1.6 million noisy labels. Their labels were noisy because tweets containing positive emoticons were labeled as positive, while those with negative emotions were labelled negative. The rest of the tweets (those without any emoticons) were discarded from the dataset. This approach aimed to achieve high results without human labelling but at the cost of using a very large dataset.

Next, we present our results for the complete classification. The best results are obtained through Support Vector Machine (SVM) applied at the second stage of the classification process. Therefore, the results below pertain only to SVM. These results use a total of two features: P (objectivity | tweet) and P (positivity | tweet). However, if we include all the features employed in the first step of the classification process, we have a list of 8 shortlisted features (3 for polarity classification and 5 for objectivity classification). The following results are reported after conducting 10-fold cross-validation:

Classes	True Positive	False Positive	Recall	Precision	F-measure
Objective	0.77	0.27	0.77	0.75	0.76
Positive	0.66	0.11	0.66	0.70	0.68
Negative	0.60	0.10	0.59	0.61	0.60
Average	0.70	0.19	0.703	0.703	0.703

Table 7: Final Results using SVM and Naive Bayes

However, when they include another portion of their data into their classification process (referred to as the HASH data), their average F-measure drops to 65%. In contrast, we achieve an average F-measure of more than 70%, showing better performance than either of these results. Additionally, we use only 8 features and 9,000 labelled tweets, while their process involves about 15 features in total and more than 220,000 tweets in their training set. Our unigram word models are also simpler than theirs, as they incorporate negation into their word models. However, like in the case of Go et al., their tweets are not labelled by humans but rather undergo noisy labelling through positive and negative emotions and hashtags.

Finally we conclude that our classification approach provides improvement in accuracy by using even the simplest features and small amount of data set. However there are still a number of things we would like to consider as future work which we mention in the next section.

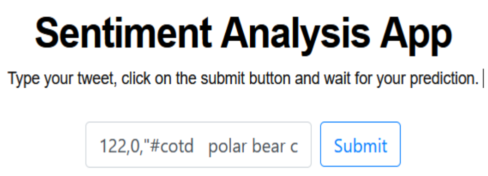


Fig 6: User input dashboard

Fig 7: Output of user-inputted comment

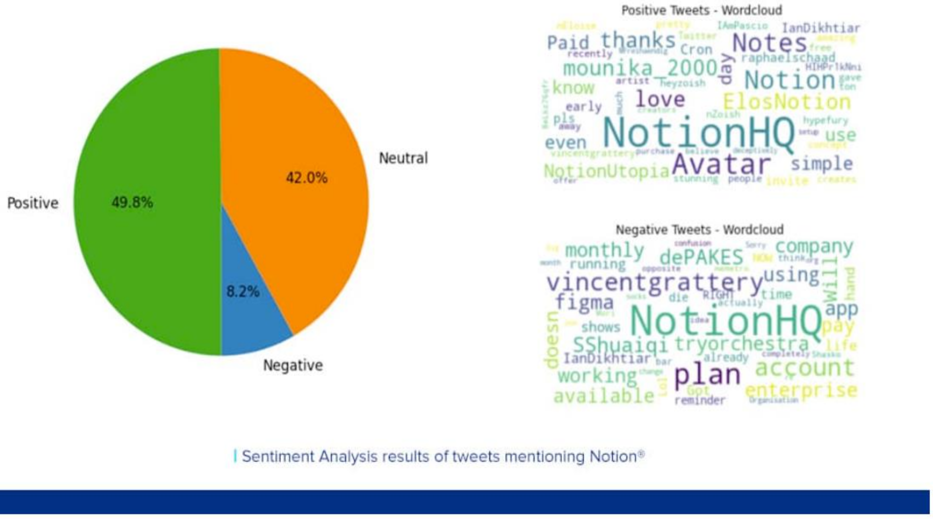
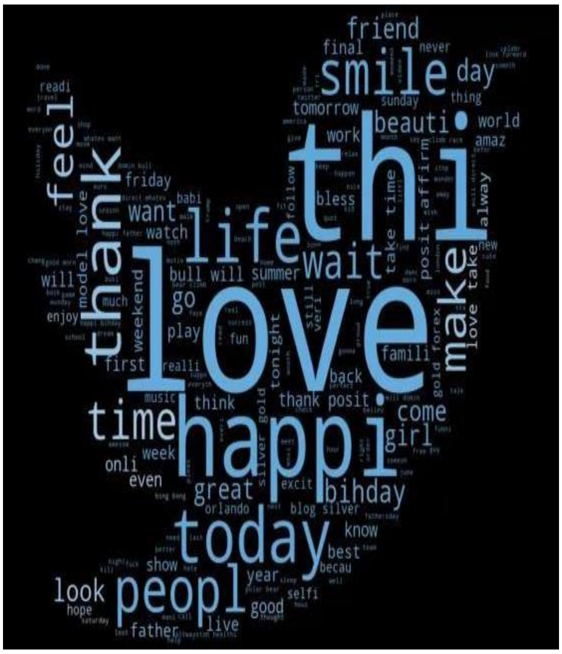


Fig 8: Pie chart class distribution



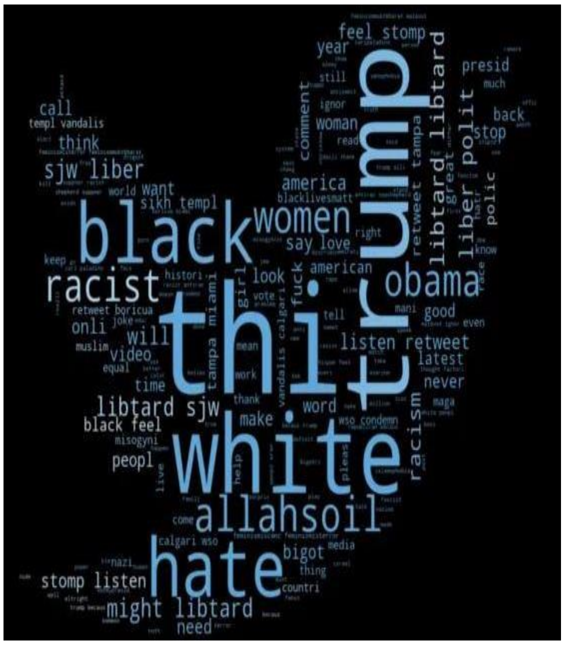


Fig 9: Data visualization of the visualization of the tweets with the label '0'

Fig 10: Image represent the data tweets with the label '1"

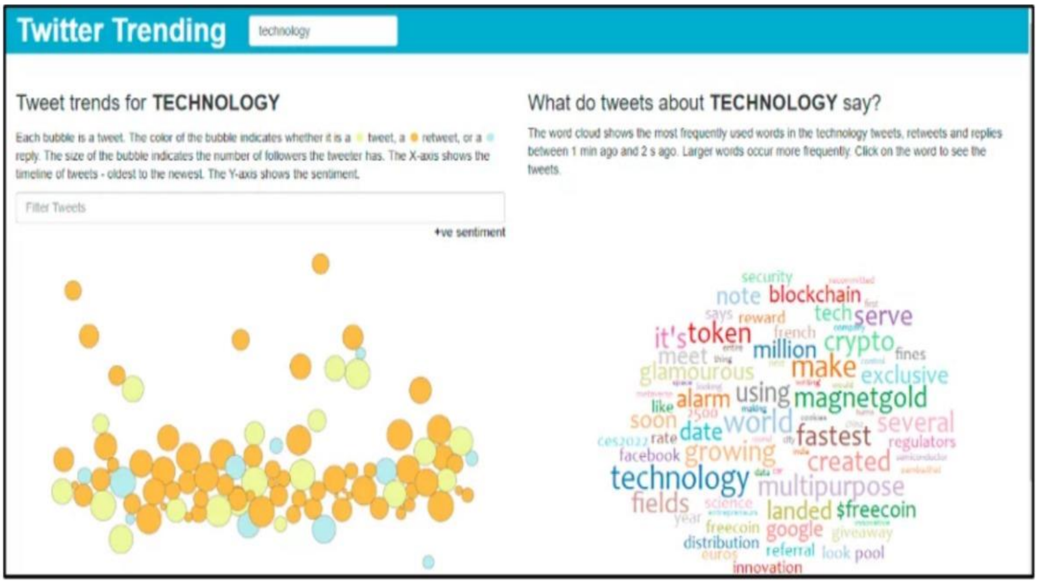


Fig 11: Analysis of Twitter trending

CHAPTER 6

SOFTWARE TESTING

Software testing is a critical component of the software development lifecycle aimed at verifying and validating that a software system meets its specified requirements and functions correctly. Here's an overview of software testing in technical terms:

1. TEST PLANNING:

This section outlines the test planning process for the weather prediction system, including test objectives, scope, and approach. It defines test scenarios, identifies testing tools and resources, and establishes timelines and responsibilities for testing activities.

2. UNIT TESTING:

It covers the unit testing phase, where individual components and modules of the system are tested in isolation to ensure their functionality and correctness. Test cases are designed to validate the behavior of specific functions, methods, and classes.

3. INTEGRATION TESTING:

This subsection focuses on integration testing, where multiple components are tested together to verify their interactions and interfaces. Integration test cases validate data flow, communication protocols, and system integration points.

4. SYSTEM TESTING:

It discusses system testing, where the entire weather prediction system is tested as a whole to evaluate its compliance with functional and non-functional requirements. Test cases simulate real-world usage scenarios to validate system behavior and performance.

5. REGRESSION TESTING:

It addresses regression testing, where previously tested components and functionalities are retested to ensure that recent changes or updates have not introduced new defects or regressions. Automated regression test suites are employed to streamline testing efforts and ensure software stability.

1. PERFORMANCE TESTING:

This part covers performance testing, where the system's scalability, reliability, and responsiveness under various load conditions are evaluated. Performance test scenarios simulate high traffic scenarios to identify bottlenecks and optimize system performance.

2. SECURITY TESTING:

It discusses security testing, where the system's resilience to security threats, vulnerabilities, and attacks is evaluated. Security test cases assess authentication mechanisms, data encryption, access controls, and protection against common security risks.

3. USER ACCEPTANCE TESTING (UAT):

This subsection focuses on user acceptance testing, where the system is tested by end-users to validate its usability, functionality, and conformance to user requirements. UAT test cases are designed to simulate real-world usage scenarios and gather user feedback.

4. ACCESSIBILITY TESTING:

It addresses accessibility testing, where the system's accessibility features and compliance with accessibility standards are evaluated. Accessibility test cases assess screen reader compatibility, keyboard navigation, and color contrast ratios to ensure inclusivity for all users.

5. USABILITY TESTING:

This part discusses usability testing, where the system's user interface and overall user experience are evaluated. Usability test cases assess navigation, layout, and interaction design to identify areas for improvement and enhance user satisfaction.

6. ERROR HANDLING AND RECOVERY TESTING:

It covers error handling and recovery testing, where the system's ability to detect, report, and recover from errors and exceptions is evaluated. Test cases simulate error scenarios to validate error messages, logging, and recovery mechanisms.

7. DOCUMENTATION AND REPORTING:

This section addresses documentation and reporting requirements for testing activities, including test plans, test cases, test results, and defect reports. Comprehensive documentation ensures traceability and transparency throughout the testing process.

Overall, this chapter provides a detailed overview of software testing activities for the weather prediction system, covering test planning, unit testing, integration testing, system testing, regression testing, performance testing, security testing, user acceptance testing, accessibility testing, usability testing, error handling and recovery testing, and documentation/reporting.

CHAPTER 7
CONCLUSION

In conclusion, sentiment analysis in the realm of micro-blogging, especially on platforms like Twitter, is an evolving field with significant potential for improvement. Our project has highlighted several key areas for future exploration and enhancement to achieve more accurate and robust sentiment analysis:

Enhancing Unigram Models: Incorporating information about the proximity of words to negation terms can improve the accuracy of sentiment analysis. Using a window approach, where the effect of negation is stronger when closer to the word, could be a promising strategy.

Exploring Bigrams and Trigrams: Utilizing higher-order n-grams as features could capture more complex patterns in language and potentially enhance classification accuracy. However, this approach would require larger labelled datasets to train the models effectively.

Integration of Parts of Speech (POS) Information: Incorporating POS information into unigram models could provide additional context and improve the classification of sentiments based on word categories. This integration may lead to more nuanced and accurate sentiment analysis.

Consideration of Word Position: Analysing the impact of the relative position of words within tweets on classification performance could reveal insights into how the structure of language influences sentiment expression.

Addressing Class Imbalance: To ensure that sentiment classifiers are not biased towards majority classes, labelling more data to achieve balance across all classes is essential. This approach would improve the overall accuracy and fairness of the classification models.

Sentiment Analysis with Partially Known Context: Focusing on specific categories like politics, celebrities, products, sports, or media could provide more targeted insights into sentiment trends within these domains.

Modelling Human Confidence: Incorporating agreement levels among human labellers into the sentiment analysis process could help quantify the certainty of sentiment classifications, leading to more reliable results.

The task of sentiment analysis, especially in the domain of micro-bloging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved performance.

Right now we have worked with only the very simplest unigram models; we can improve those models by adding extra information like closeness of the word with a negation word. We could specify a window prior to the word (a window could for example be of 2 or 3 words) under consideration and the effect of negation may be incorporated into the model if it lies within that window. The closer the negation word is to the unigram word whose prior polarity is to be calculated, the more it should affect the polarity. For example if the negation is right next to the word, it may simply reverse the polarity of that word and farther the negation is from the word the more minimized ifs effect should be.

Apart from this, we are currently only focusing on unigrams and the effect of bigrams and trigrams may be explored. As reported in the literature review section when bigrams are used along with unigrams this usually enhances performance.

However for bigrams and trigrams to be an effective feature we need a much more labeled data set than our meager 9,000 tweets.

Right now we are exploring Parts of Speech separate from the unigram models, we can try to incorporate POS information within our unigram models in future. So say instead of calculating a single probability for each word like *P (word | obj)* we could instead have multiple probabilities for each according to the Part of Speech the word belongs to. For example we may have *P (word | obj, verb), P (word | obj, noun) and P (word | obj, adjective).* Pang et al. [5] used a somewhat similar approach and claims that appending POS information for every unigram results in no significant change in performance (with Naive Bayes performing slightly better and SVM having a slight decrease in performance), while there is a significant decrease in accuracy if only adjective unigrams are used as features. However these results are for classification of reviews and may be verified for sentiment analysis on micro blogging websites like Twitter.

One more feature we that is worth exploring is whether the information about relative position of word in a tweet has any effect on the performance of the classifier. Although Pang et al. explored a similar feature and reported negative results, their results were based on reviews which are very different from tweets and they worked on an extremely simple model.

One potential problem with our research is that the sizes of the three classes are not equal. The objective class which contains 4,543 tweets is about twice the sizes of positive and negative classes which contain 2,543 and 1,877 tweets respectively. The problem with unequal classes is that the classifier tries to increase the overall accuracy of the system by increasing the accuracy of the majority class, even if that comes at the cost of decrease in accuracy of the minority classes. That is the very reason why we report significantly higher accuracies for objective class as opposed to positive or negative classes. To overcome this problem and have the classifier exhibit no bias towards any of the classes, it is necessary to label more data (tweets) so that all three of our classes are almost equal.

In this research we are focussing on general sentiment analysis. There is potential of work in the field of sentiment analysis with partially known context. For example we noticed that users generally use our website for specific types of keywords which can divided into a couple of distinct classes, namely: politics/politicians, celebrities, products/brands, sports/sportsmen, media/movies/music. So we can attempt to perform separate sentiment analysis on tweets that only belong to one of these classes (i.e. the training data would not be general but specific to one of these categories) and compare the results we get if we apply general sentiment analysis on it instead.

Last but not the least, we can attempt to model human confidence in our system. For example if we have 5 human labellers labelling each tweet, we can plot the tweet in the 2-dimensional objectivity / subjectivity and positivity / negativity plane while differentiating between tweets in which all 5 labels agree, only 4 agree, only 3 agree or no majority vote is reached. We could develop our custom cost function for coming up with optimized class boundaries such that highest weightage is given to those tweets in which all 5 labels agree and as the number of agreements start decreasing, so do the weights assigned. In this way the effects of human confidence can be visualized in sentiment analysis.

In conclusion, our project contributes to the ongoing development of sentiment analysis techniques for social media data, particularly in the context of the 2019 Indian General Elections. By exploring various machine learning models and feature extraction techniques, we have demonstrated the potential for accurate sentiment classification. Moving forward, further research in the aforementioned areas could lead to significant advancements in sentiment analysis on micro-blogging platforms, improving our understanding of public opinion and sentiment trends.

APPENDICES

REFERENCES

1. Wong, F. M. (2013). Quantifying Political Leaning from Tweets and Retweets. ICWSM.

2. Johann Bollen, Alberto Pepe and Huia Mao. Modelling Public Mood and Emotion: Twitter Sentiment and socio-economic phenomena. *In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.

3. Boutet, A. K. (2012). What's in your Tweets? I know who you supported in the UK 2010 general election. Proceedings of the International AAAI Conference on Weblogs and Social Media.

4. Golbeck, J. &. (2011). Computing political preference among twitter followers. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.

5. Pennacchiotti, M. &. (2011). Democrats, republicans and Starbucks aficionados: user classification in twitter. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 430-438.

6. Saif, Hassan & Fernandez, Miriam & Alani, Harith. (2013). Evaluation Datasets for Twitter Sentiment Analysis. A survey and a new dataset, the STS-Gold. CEUR Workshop Proceedings.

Websites:

1. https://www.irjet.net/archives/V8/i3/IRJET-V8I3355.pdf
2. https://github.com/berkgulay/sentiment-analysis-from-image
3. http://www.w3school.com
4. https://www.geeksforgeeks.org/sentiment-analysis-on-social-media-project-check-today-tweet-for-analysis/