

FACULTY OF SCIENCE, ENGINEERING AND COMPUTING

School of Science, Engineering, and Computing

MSc DEGREE

in

Data Science

Harshita Kakadiya Maheshbhai

KU Number: K2172452

CI7000: Project Dissertation

Project Title: Sentiment Analysis Using Machine Learning

Supervisor: Nada Philip

Kingston University London

TABLE OF CONTENTS

List of Figures	Error! Bookmark not defined.
List of Abbreviations	Error! Bookmark not defined.
List of Figures	3
List of Abbreviations	3
Abstract	4
Acknowledgement	5
Chapter 1: Introduction	6
1.1 Background and Motivation	6
1.2 Aims and Objectives	7
1.3 Ethics and Legal Relevance	8
Chapter 2: Initial Literature Review	9
Chapter 3: Methodology	12
3.1 Levels of Sentiment Analysis:	12
3.2 Sentiment Analysis Methods:	14
3.3 Techniques for sentiment analysis	19
3.4 Software Tools Used for sentiment analysis:.....	20
3.5 Libraries for Sentiment Analysis:	21
Chapter 4: Data Collection and Data Analysis	22
4.1 Data Collection	22
4.2 Data Analysis:	23
Chapter 5: Designing and Implementation	26
5.1 Data Collection	26
5.2 Preprocessing of the data:	27
5.3 Extracting Feature:	30
5.4 Feature Selection:.....	31
5.5 Training and testing the model	33
Chapter 6: Model Performance Evaluation Metrics	37
Chapter 7: Conclusion.....	41
Chapter 8: Future work	42
Reference	42

List of Figures

Figure 1: Overview of the Dataset	23
Figure 2: Visualize the Number of Positive and Negative Sentiment	24
Figure 3: Visualize the Most Common Positive Review word.....	24
Figure 4: Visualize the Most Common Negative Review word	25
Figure 5: Count the Number of words in Sentence.....	25
Figure 6: Load the data set.....	26
Figure 7: Summarization of the Data Frame.....	27
Figure 8: Counts of the distinct Value	27
Figure 9: Apply Data Pre-processing.....	28
Figure 10: Performing Stemming and Lemmatization	30
Figure 11: Performing Feature Extraction	31
Figure 12: Split the Data into Training and Testing	33
Figure 13: Logistic Regression Model.....	34
Figure 14: Multinomial Naïve Bayes Model	34
Figure 15: Support Vector Machine Model	35
Figure 16: Long Short Term Memory Model	36
Figure 17: Confusion Matrix for Logistic Regression.....	38
Figure 18: Confusion Matrix for Multinomial Naïve Bayes	39
Figure 19: Confusion Matrix for Support Vector Machine	40

List of Abbreviations

SA	Sentiment Analysis
ML	Machine Learning
EDA	Exploratory Data Analysis
ANN	Artificial Neural Network
NLTK	Natural language Toolkit
SVM	Support vector Machines
RNN	Recurrent Neural Network
KNN	K-Nearest Neighbours
DT	Decision Tree
DNN	Deep Neural Network
LSTM	Long Short-Term memory
CV	Cross validation
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
TF-IDF	Term Frequency – Inverse Document Frequency
UI	User Interface
BoW	bag-of-words

Abstract

It is difficult to keep up with all the developments in the field of sentiment analysis, one of the computer science study fields that are expanding the quickest. Natural language processing is a field that studies the expression of sentiment in free text using sentiment analysis. As a result of Sentiment Analysis, valuable information can be derived from textual data to identify the emotional tone behind a statement.

In a variety of social situations, including marketing, economics, and politics, it has found useful uses. As stated in the definition of health, which is "a condition of full physical, mental, and social well-being and not only the absence of sickness or disability," this evaluation focuses especially on applications connected to health.

There are many ways to do sentiment analysis. Machine Learning Classifiers were used to do Sentiment Analysis. Users' attitudes are classified as either "good" or "negative" using deep learning models.

This study tries to develop a classifier that can predict human behavior based on a person's mood using the Sentiment Reviews dataset. Naive Bayes, Support Vector Machine, Decision Tree, and Random Forest are some examples of text classification algorithms. These algorithms use various feature extraction methods, including Bag of Words and TF-IDF. The models are further assessed using measures for accuracy, precision, recall, and F-score. Based on our experiments, we found that Logistic Regression produces the best results and is extremely useful for identifying human sentiments.

Acknowledgement

To begin with, I would like to express my sincere gratitude to my supervisor, Professor Dr. Nada Philip. It was an absolute pleasure working with her from the beginning to the end, and I am so grateful for all of her suggestions and guidance.

It has been a pleasure working with my colleagues, and I particularly appreciate their advice, assistance, and valuable insights in some areas of difficulty. A great deal of this research paper's success can also be attributed to the advice and support provided by those people.

I would especially want to thank my parents for their encouragement and support, which they have given me, on behalf of my family.

Last but not least, I want to thank the University of Kingston, particularly the Department of Data Science, for all of their help and the many services they have offered.

Chapter 1: Introduction

1.1 Background and Motivation

Many people today express their thoughts, feelings, and experiences through social networks and the Internet.

As a part of sentiment analysis, or "opinion mining," natural language processing, text analysis, and computational linguistics are used to automatically classify sentiment expressed in free text.

The Internet facilitates the exchange of a great deal of information, which has necessitated sentiment analysis in order to extract useful information. As a concept, sentiment analysis was first introduced by Nasukawa (Nasukawa and Yi, 2003). In the natural language processing process, sentiment analysis (NLP) (Hussein, Doaa Mohey El-Din Mohamed, 2018) begins with analysing the opinions, feelings, and reactions of users on social networking sites and business websites regarding the Numerous goods and services are offered online.

Opinion mining is sometimes referred to as value classification, which aids in classifying thoughts and views into positive, negative, or neutral categories. This technique is like sentiment analysis. The many forms of emotions are also discoverable. Sentiment analysis is a technique used to analyze text-based resources such as social media posts and online reviews.

We have recently had a lot of fruitless conversations with friends, co-workers, family members, and other individuals. Occasionally, we say something incorrectly and later regret it. There are some people who suffer from mental illness that can lead to despair, mental health disorders, and other severe issues. When someone is alone, the possibility of them being sad increases. The conversation doesn't have any recommendations for improving it. As a solution, we propose developing a web application that may assist in relationship development and, as a result, lower the risk of mental illness.

The two languages on which most sentiment analysis research has been concentrated are English and Chinese. There are now just a few scholars working on study in other languages, including as Arabic, Italian, and Thai. This research uses a number of techniques to several datasets, including movie reviews and product reviews, and it spans the years 2004 to the present (Jagdale, Shirsat and Deshmukh, 2016).

The use of sentiment analysis techniques like natural language processing, statistics, and machine learning is highly efficient for identifying sentiment within text units. Sentiment analysis has found practical applications in social domains such as marketing, economics, and politics.

Applications related to health are the main focus of this evaluation.

Sentiment analysis in this field can be challenging because society tends to focus on negative events like illnesses, accidents, and impairments when it comes to health issues. The quality of life of a patient with a chronic illness depends on how well those symptoms are controlled and treated rather than whether the patient has accompanying symptoms. As a result of the negative connotation that health problems have, sentiment analysis findings tend to favor the negative spectrum.

There are many distinct types of challenges in sentiment analysis. The word "one" is a phrase of opinion that, depending on the situation, may be good in one situation and negative in another. Every time individuals don't express their thoughts in the same manner, it creates a new issue. For the bulk of traditional text processing, the distinction between two-word fragments serves as the foundation. But in terms of sentiment analysis, "the image was gorgeous" and "the picture was not pleasant" are quite different.

Contradictory opinions may exist among individuals. People say both positive and bad things, and you can handle this to some extent by looking at each comment independently. People offer their thoughts on a number of unofficial venues, such as Twitter, blogs, Facebook, Amazon, etc.

It finds out whether customers approve or disapprove of a product based on their comments and reviews on commercial websites; this helps to increase sales for the business since it reveals the preferences of a client. In response to the influx of various viewpoints on social networking sites, systems, politicians, psychologists, manufacturers, and researchers developed new theories to examine the data and make the best possible judgments. Using sentiment analysis techniques such as NLP, statistics, and machine learning, it is possible to extract and define sentiment information in a text unit.

Although something can be read by people, computers have a hard time understanding it. Even other individuals may sometimes struggle to understand what someone was thinking when a quick message lacks context. It all depends on what the person meant when they said, "That movie was as wonderful as its earlier movie," for example.

In terms of business intelligence, sentiment analysis is used in a variety of ways. For instance, in marketing, it may be used to evaluate the success of a promotional campaign or the launch of a new product, identify the most well-liked versions of a certain item or service, and even identify which demographics like or disapprove of a specific feature.

1.2 Aims and Objectives

Aims:

The main goal is to classify the emotions expressed by phrases or words in free-form text using Machine Learning.

Objectives:

An objective of sentiment analysis is to categorize people's opinions into positive, negative, or neutral sentiments using many unstructured review texts. From the Kaggle website, you can

download the Sentiment-140 data set in CSV format. Approximately 1,600,000 tweets are included in it, which were pulled from the Twitter API. We can detect sentiment based on the annotated tweets (0 = negative, 1 = positive).

- **In order to perform Data Pre-processing and Exploratory Data Analysis (EDA):**

Visualizing and analysing the data is part of the EDA process, which aims to determine the most significant data properties. In addition to identifying obvious errors, it can also help identify patterns within the data, detect outliers, and discover interesting relationships between variables. This will be accomplished using Python Notebooks (a Google collaboration) so that it will be easier to read and understand. It will be necessary to use libraries such as Pandas, NumPy, Sea-borna, and Matplotlib.

Pre-processing is necessary for us to be able to train models on our dataset. As part of this section, Pre-processing steps will be performed such as dropping and filling missing values, transforming data, generating features, and encoding features.

- **Various libraries, including NLTK and Word cloud, were used to analyse sentiment.**
- **A machine learning model needs to be trained and evaluated on the data in order to accomplish this goal**

The most well-liked machine learning libraries for Python are scikit-learn and deep learning utilizing Keras (TensorFlow), and they will be utilized in this project's modeling section. Support vector machines, naive Bayes, linear regression, and long short-term memory are examples of machine learning algorithms. Long short-term memory is a deep learning architecture based on an artificial recurrent neural network. to make predictions by teaching our model. Metrics like as accuracy, F1 score, confusion matrix, and other parameters will be used to evaluate the model's performance.

- **To build User Interface**

To build attractive, natively built apps for mobile, web, and desktop from a single codebase, use Google's portable UI toolkit called Flutter.

1.3 Ethics and Legal Relevance

For several reasons, research ethics are crucial. However, the use of social media in the context of disclosing personal or private information may raise ethical issues, such as those relating to privacy and confidentiality. The need to communicate with patients online while strictly adhering to data privacy laws has led to an increase in the number of websites and networks created specifically to provide a safe environment for sharing health-related information online. As a result, I'll make use of an online, open-source dataset. Thus, I'll make use of a dataset that is presently available online and is open source. I am aware of each and every one of these considerations, and I will make sure to honor them all when I carry out my study.

Chapter 2: Initial Literature Review

This chapter will cover the literature on the topic, including the many approaches used in sentiment analysis, as well as their findings, conclusions, and shortcomings.

(Pang, Lee and Vaithyanathan, 2002) Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan's study pertains to document-level sentiment analysis on a collection of movie review data. The authors employed a range of machine learning techniques to make their classifications, such as Naive Bayes, maximum entropy classification, and support vector machines. It was found on the IMDB website that the data used for this research came from. A website called IMDB provides information about films, television programs, home videos, video games, and online streaming entertainment - including cast, crew, and personal biographies, plot summaries, trivia, ratings, and reviews from movie fans and critics as well as information on casting, crew, and personal biographies. Reviews that were rated using stars or another numeric number were only taken into consideration by these researchers. A wide variety of 1-grams, 2-grams, unigrams plus parts of speech, adjectives, the top 2,633 unigrams, and the unigrams plus position on the list were included. The results of the study were surprising compared to what we would have expected based on human measurements. Support Vector Machine is an algorithm that is the most effective according to a comparison with other classifiers.

(Turney, 2002) Using unsupervised learning techniques, Peter David Turney presents a method that allows him to classify a review as either recommended or not recommended. In order to classify reviews according to their semantic orientation, adjectives and adverbs are included in the phrases. The purpose of this research is to analyze sentiment at the level of the document using sentiment analysis. By computing point-wise mutual information, it is possible to calculate (PMI) the semantic orientation of the phrase and the word. The company conducted surveys for a variety of businesses, including automobiles, banks, movies, tourist attractions, and destinations that cater to the automotive industry. They obtained accuracy scores ranging from 84% for automobile reviews to 66% for film reviews.

The authors of this study (Hatzivassiloglou and McKeown) developed a computational method that predicts semantic orientation in 1997 with Vasileios Hatzivassiloglou and Kathleen McKeown. During the algorithm's development, adjectives and adverbs were not the focus of the algorithm's development. There is a strong emphasis on singular adjectives in the following section. An approach based on conjunction limitations was applied to identify the semantic orientation of adjectives using a four-step supervised learning approach. There was an accuracy range of 78% to 92% for categorizing adjectives. This was depending on how much training data they had and the amount of data they had at their disposal.

(Tong, 2001) R. M. Tong developed a strategy for generating emotion timelines. Using an algorithm, it analyzes online movie reviews over a period of time, integrating both positive and negative signals to build a narrative of a movie. There was a domain vocabulary used by them for movies. A lexicon that is generated automatically is used instead of one compiled manually. A significant number of applications for this work include deploying automated review scoring, monitoring marketing activity, analyzing voter sentiment for politicians, analyzing stock traders' financial viewpoints, and analyzing trends in entertainment and technology based on the opinions of trend analysts.

In terms of Janyce M. Wiebe's work, she describes it as co-constructed subjective labelling (Wiebe, 2000). Their evaluation of performance was objective. A method has been developed in this study in order to classify words based on distributional similarity in order to identify significant subjective markers. The findings of 10-fold cross-validation indicate that features based on both similarity clusters and lexical semantic characteristics are more precise than features built on either one alone.

It has been demonstrated that Pang, Bo, and Lillian Lee employ a multi-way classification technique to categorise the polarity of a document, as well as to expand the task of categorizing comments about movies as being negative or positive in order to predict their star ratings on a scale of 3, a device described in Pang, Bo, and Lillian Lee's new study (Pang and Lee, 2005). As a result, they evaluated the degree to which individuals were successful at their jobs. An algorithm based on the labelling of metrics is used in order to develop the meta-algorithm. When a situation-appropriate similarity metric is applied, it can determine whether the meta approach outperforms either the multi-class or regression versions of Support Vector Machines. Among the datasets they used were movie reviews from a variety of sources.

The thoughts or feelings indicated on various elements or facets of things, such as a smartphone, a digital camera, or a bank, were ascertained by Mining Hu and Bing Liu.

Aspects of several entities are examined in detail.

This article accomplished three things.

- (1) identifying product characteristics that consumers have mentioned
- (2) determining whether each review's opinion sentences are good or negative by recognising them.
- (3) writing an outcomes summary.

A number of techniques were proposed in the proposal by the authors, such as Part-Of-Speech tagging, Recognition of Frequently Used Characteristics, Object extraction from opinions, Sentiment Analysis Orientation Detection, Identification of Infrequent Features, Identification of Viewpoints in Opinion Sentences, and Summary Generation. In terms of accuracy and recall, the average accuracy and recall for opinion sentence extraction for five commodities is 0.64 percent and 0.69 percent, respectively. A prediction of sentence orientation based on opinion extraction and opinion extraction is derived from the results of the analysis. According to their study, the sentence orientation accuracy was 0.84 percent. They achieved 0.84 percent accuracy in sentence orientation.

A study by Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu used data from blogs to develop a two-class classification problem for estimating sales performance. Amounts collected from IMDB's website include one month's worth of box office figures (daily gross revenue) for each film for the previous month. PLSA (Probabilistic Latent Semantic Analysis) is a tool for sentiment analysis that the company offers. It was proposed that they use ARSA: A SENTIMENT-AWARE MODEL for the purpose of forecasting product sales based on data collected from blogs on sentiment.

According to Wilson, Wiebe, and Hoffman, using a machine learning approach, they propose a strategy for determining the polarity or neutrality of a phrase within a paragraph. Using this approach, it is possible to improve the classification of emotion polarity by taking into

consideration phrase-level context, for example, whether the emotion transmitted is modified or negated by an adverb.

According to Seongik Park, in order to strengthen the reliability of the lexicon, thesaurus items are compiled from seed words that are taken from three online dictionaries that have a wide vocabulary. A vocabulary for categorizing emotions could then be built using this technique in order to build a vocabulary. An annotated thesaurus is a collection of synonyms and antonyms that help extend or expand the vocabulary of a sentence. In light of the fact that he focused primarily on creating lexicons, the product did not undergo many changes.

The literature review serves as a reminder of the research that has been done in the past. We conducted this survey in order to determine the state of health and well-being of the participants. After each contact, this method is designed to deliver automated feedback to enhance future engagements, with the purpose of improving future contact. Besides employing a chat mechanism, it also embeds a call analysis mechanism that allows us to analyze what they are feeling via their call recordings. Moreover, it is designed with an AI bot. this chatbot users can use to express concerns, get recommendations, and obtain answers with the help of this chatbot when they are out and about.

Chapter 3: Methodology

3.1 Levels of Sentiment Analysis:

We examined trends on the basis of documents, sentences, phrases, and aspects. Identify and analyse sentiment at each level of the document including document level, phrase level, and aspect level. The implementation of text data may be accomplished in a variety of ways, depending on its magnitude and complexity.

1. Documents-level Sentiment Analysis
2. Sentence-Level Sentiment Analysis
3. Word/Phrase Level Sentiment Analysis
4. Aspect Level Word/Phrase Level Sentiment Analysis

1. Documents-level Sentiment Analysis:

At the very beginning, SA is determined by the level of data that is available. An aspect of a document is assigned by means of sentiment classification that is based on a data set of sentiments. The use of sentiment analysis in this form is not common. As a result, it is classified, if any, in accordance with the tone of the overall work in which it has been included. A good or negative rating can be applied to a chapter or section of an author's work in order to categorize it. For categorizing the material at this stage, both supervised and unsupervised learning methods can be used. In this instance, it is often presumed that the opinion holder is a single person or source [12]. Sentiment regression is an additional concern with SA at the document level [6, 25-28]. It has also been reported that some researchers have developed a learning algorithm to predict an article's evaluation results based on the degree to which the article was positively or negatively evaluated [6]. A linear-based combination method can be found in [3, 12] for analysing the polarities of a text document using a linear-based combination method. Having specialized in only writing documents at this point, one of the biggest challenges is that verbs used to express opinions in the text may not be regarded as subjectivity statements at this stage since they are simply verbs that express opinions. More precise results can be obtained from SA if each phrase is analysed separately in order to obtain more accurate results. Due to this, only objective statements must be rejected, although subjective sentences must be retrieved for sentiment analysis, which is carried out on them. As a consequence, SA research at the phrase level has been one of the most prominent research priorities in recent years. Therefore, it has been a prominent priority to conduct research on SA at the phrase level

2. Sentence-Level Sentiment Analysis

At this point in the assessment, each phrase is analyzed in terms of its polarity and its significance is determined. I would highly recommend this method when it comes to writing papers that are filled with many different emotions (Yang and Cardie, 2014). In Rao et al. (2018), a subjective level of categorization can be found at this level of categorization. Using the same techniques and more training data as those used at the

document level, each sentence's polarity will be calculated individually. This is done using the same techniques as those used at the document level. Individually or collectively, the polarity of each phrase may be utilised to determine the emotion of the text. Sometimes, document-level sentiment analysis is inadequate for certain applications (Behdenna et al. 2018). The challenge comes when the tasks are a bit more complex, such as dealing with conditional phrases and ambiguous assertions (Ferrari and Esuli, 2019). It is imperative under these circumstances to be able to classify the sentiment at the sentence level.

In addition, sentiment analysis at the sentence level has limitations as well. The word "feelings" may not appear in some objective phrases while it is actually referenced in others. In this instance, I would like to point out that the cup I purchased a week ago has developed fractures on the sides, and they are quite visible. As it conveys facts, the previous clause is a suitable example of an objective clause since it conveys information that is true. After a deep examination, it seems there is an indirect message in this line. Regarding the fractures on the side of the cup, the speaker of the opinion expressed a negative opinion about the condition of the cup. Using sentiment analysis at the level of individual words or phrases is the most effective way to solve this problem. An explanation of how sentiment analysis can be applied at the word level follows.

3. Word/Phrase Level Sentiment Analysis

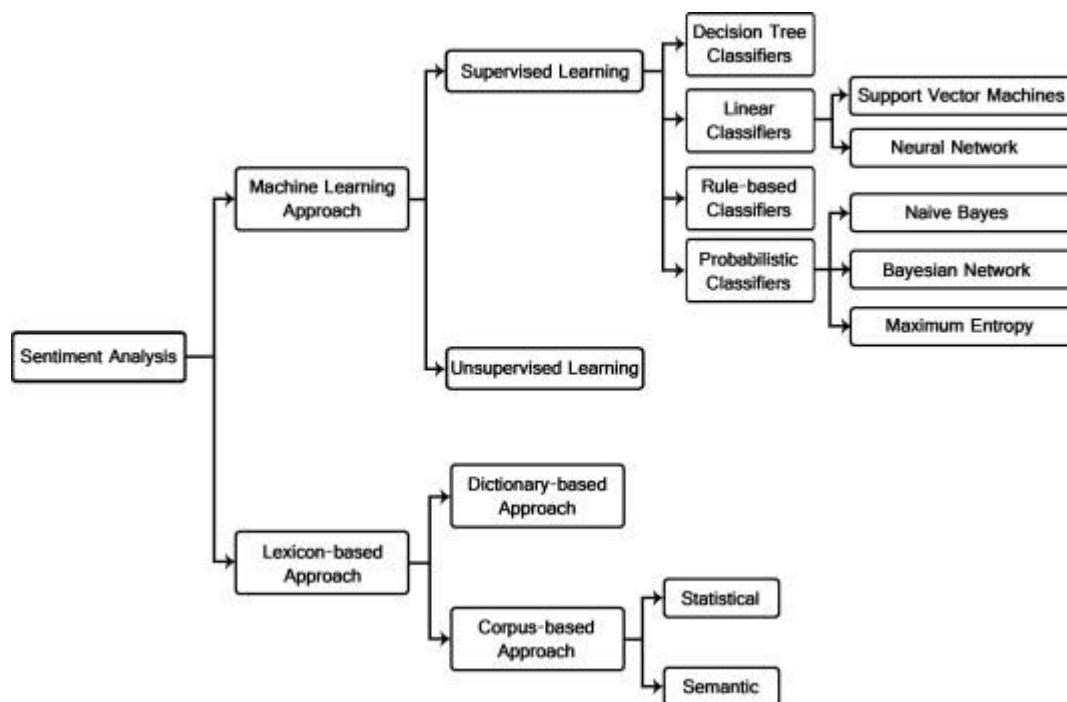
It is essential to keep in mind the specific words or phrases in question when doing sentiment analysis (SA) at the level of individual words or phrases. A word is one of the smallest meaningful units that make up a written piece. Therefore, being able to recognize a word in its smallest meaningful unit is very helpful. Due to the nature of this type of SA, the most information is gained. Due to its technical advantages, SA has attracted the attention of a considerable number of scholars in the last few decades. This explains why the number of research papers pertaining to SA at the word level is simply innumerable as a result of this. A sentiment categorization system at the sentence and document levels is implemented using the polarity of phrases and words, which has been taken into account in earlier research studies. It is therefore not uncommon for word lexicon lists to be produced both manually and mechanically as a direct consequence of this. There is a general consensus that the vocabulary of SA contains a variety of adjectives (such as gorgeous, pleasant, beautiful, amazing, old, awful, and dreadful), as well as adverbs (e.g. quickly, slowly, badly, and horrifically), and certain verbs (including like, hate, love, and detest). A noun like "garbage" or "junk" may be taken by some to mean something that is emotionally charged as opposed to simply being a noun.

4. Aspect Level Word/Phrase Level Sentiment Analysis

Sentiment analysis is performed at an aspect level to analyse sentiments. The aspect-based method of sentiment analysis takes into account the possibility that any particular statement may have multiple facets to it. The key to understanding the sentence is to focus on all of the aspects that are used in it. Using the polarity as a guideline, give each facet of the phrase a polarity, and you will have created a single emotion for the phrase. A feature-based, a topic-based, an entity-based, or a target-based analysis will focus on characteristics,

including the words related to the emotions, rather than linguistic structures (documents, paragraphs, sentences, clauses, phrases, or paraphrases). Three characteristics define an entity, and the words related to those characteristics define sentiment. The first step in the analysis process is to determine exactly what is being observed (the "thing under observation"). When determining and classifying the emotions associated with an entity (positive, negative, and neutral), polarity is used to determine and classify the emotions associated with that entity. As a result, the entity's facets are vividly discovered and scored as per their polarity, thus enabling a finer grain examination to happen as a result of the entity's facets being vividly discovered and scored based on their polarity.

3.2 Sentiment Analysis Methods:



In the data mining process, different types of sentiment analysis methods can be distinguished in the literature,

- (1) machine learning,
- (2) rule-/dictionary-based, and
- (3) hybrid approaches

1. Machine learning

Using machine learning algorithms, we are able to classify feelings in ways that are useful to us. Sentiment analysis is the process of identifying and measuring the attitude conveyed in an article of writing or a speech through the analysis of the language used. It is a multi-step process in which techniques such as natural language processing, text analysis, computational linguistics, and so on are used together with other methods.

There are two primary in Machine Learning approaches to sentiment analysis:

1. Supervised machine learning
2. Lexicon-based unsupervised learning

1. Supervised machine learning

Machine Learning with Supervision This method is used by the algorithm to learn. Based on a sample of the training data the algorithm makes predictions about the result in real-time until it reaches a level of performance that is deemed satisfactory. It is already known what the result will be before the event even takes place. In many cases, supervised learning is used for mapping categorical input data into labelled classes or for performing regression analyses on the data [11]. Unlike discrete output data, continuous output data is generated as a result of converting input data into continuous output. To achieve this goal, we need to recognize and predict the correct conclusion based on a particular correlation between the input information and the outcome.

As we will be dealing with real-world events in the future, the data will not have labels that have been predetermined prior to the data's analysis, which is a requirement for supervised machine learning approaches [8]. This is due to the fact that real-world scenarios are unpredictable in nature. A machine learning algorithm based on unsupervised learning is used in this scenario to categorise the data on the basis of some feature similarity. The models derived from the class fresh data are then used for predicting classes in the future. It is the purpose of this endeavour to determine the natural structure or distribution of data points within a collection of data points. This is without the use of labels that have been specifically provided for the purpose of doing so. In the case of unsupervised learning methods, comparing the performance of different models is challenging. This is because there are no labels provided to help in comparing the performance of different models. It is imperative, however, that this scale be useful for exploration and dimensional reduction.

Some supervised algorithms are as follows:

- Support Vector Machines (SVM)
- Naive Bayes (NB)
- Logistic Regression (LogR)
- Maximum Entropy (ME)
- K-Nearest Neighbor (kNN)
- Random Forest (RF)
- Decision Trees (DT)

There are three types of approaches:

1. Lexicon based Methods
2. Dictionary Based Methods
3. Corpus Based Methods

1. Lexicon based Methods

Each lexicon consists of a series of tokens, each of which carries a score that indicates whether the given text is neutral, positive, or negative depending on its score (Kiritchenko et al. 2014). The tokens are assigned a score based on their polarity, such as +1, 0, -1 depending on whether or not they are positive, neutral, or negative. There are also certain scores based on the strength of the polarity. A score of +1 will indicate an extremely high level of positive valence while a score of -1 will indicate a very low level of valence. Lexicon-based approaches rely on scoring tokens within a review or text, and these scores are aggregated to develop a score for each token, i.e., good, bad, and unbiased rating tokens are added independently to create a score. As a final step, the sentence is given an overall polarity by combining the scores of each of the sentences individually. It is first decided to divide the text into tokens that comprise single words, and then it will be determined that each token is polar and aggregated based on its polarity.

As lexicon-assisted sentiment analysis is able to analyze both phrase and feature level sentiments, it can be very useful. In this approach, there is no need for training data since no training data will be used, making it an unsupervised approach. Considering how words are able to possess several meanings and senses depending on the context, a positive term in one domain may be negative in another. In this way, the main disadvantage of this method is that it is domain-specific. Taking the word "small" and paired it with the sentences "The TV screen is too small" and "This camera is extremely small," we can see that in the first sentence the word "small" is negative, as people prefer larger screens, whereas in the second sentence it is positive, since it will be more convenient to carry the camera if it is small. The problem is best solved by developing a domain-specific sentiment lexicon or by adapting a language already in use.

In addition to the obvious advantages of lexicon-based approaches, one of them is that they do not require any training data and, by some experts, they are considered to be an unsupervised approach (Yan-Yan et al. 2010). There is one primary disadvantage of the lexicon-based method, and it is the fact that it is largely domain-specific, which means that it cannot be used in a domain that is clearly distinct from the one where the terms are used (Moreo et al. 2012). In the case of the word enormous, it may work well or poorly, depending on the context in which it is used. There is a possibility that "there was an enormous delay in the network" will be perceived as a good case in comparison with "the wait for the movie was enormous," a term that may be seen as a negative case. The domain should therefore be carefully considered when assigning polarities to words, as it should provide a clear understanding of their meanings. It should be noted that Table 3, displaying the comparative analysis of the Lexicon-Based Classification Method and Its Individual Advantages and Disadvantages, demonstrates the individual advantages and disadvantages of each method. Following is a description of two general methods of data analysis: corpus-based and statistical.

2. Corpus based approach

This approach employs a combination of both semantic and syntactic patterns in order to identify the emotional content of a sentence. A pre-determined set of sentiment words and their orientation is set up for the strategy. A large corpus is then analysed to identify its sentiment tokens and to discover their orientations by analysing syntactic or contextual patterns. The approach needed to construct this machine learning system is situation-specific and requires a

sufficient amount of labeled data for training. It includes, however, a method for resolving the problem of opinion words that differ according to context.

In order to analyze sentiment, Park & Kim (2016) used a corpus-based method based on the analysis of text corpora. By using language limitations and connectives, they were able to determine what the newly created token meant and what its purpose was. A correlative conjunction such as "AND" tends to have tokens with similar orientations on each side, whereas terms such as "OR" indicate shifts in opinion or tokens with different orientations on each side of the conjunction. There are times when sentiment consistency is stated as if it were a level of consistency, yet in reality, it is not in fact quite that simple. The data was collected by constructing a graph that would have tokens as vertices and their connected words as edges. Based on this graph, a linear log model would be used to determine whether two conjoined adjectives are vertical, horizontal, or polar. This would then be grouped into positive or negative sentences.

The corpus-based approach has the following types of approaches:

1. Statistical Approach and
2. Semantic Approach

1. Statistical Approach

There are several statistical techniques that can be used to identify seed opinion words or co-occurrence patterns using seed opinion words. The basic premise behind this approach is that if a term occurs more often in positive texts as opposed than negative texts, it is more likely to be positive. And the same holds true for the opposite. According to this method's central principle, similar emotion tokens will likely have the same orientation as they are detected regularly in the same environment in the same situations.

It follows that the orientation of the new token is determined by the frequency with which it appears within the context in which the existing tokens occur.

It is possible to identify seed opinion words or co-occurrence patterns using statistical techniques. As a default strategy, this approach assumes that if a term occurs more frequently in positive texts than in negative texts, it is more likely that it is a positive term, and vice versa if it is a negative term. According to the central principle of this method, if similar emotion tokens are detected frequently in the same environment, it is likely that they will have the same direction of orientation, as they indicate the same sentiment.

Therefore, it should be noted that the orientation of the upcoming token will determine its existence in accordance with the frequency with which it will occur. This will be in the same context as existing tokens. This study validated the findings by analysing data from Amazon.com's book reviews.

2. Semantic Approach:

The similarity score between tokens analysed for sentiment analysis is produced by this method in order to determine similarity scores. This is one of the most common reasons for making use of WordNet. The advantage of this method is that people are able to locate antonyms or

synonyms easily. This is because words with comparable meanings have higher scores or a higher value. This makes finding them easy. With the lexical model used in conjunction with the semantic method, a lexical model can be developed that can help define adjectives, verbs, and nouns in Sentiment Analysis. This is depending on the application. Throughout the length of the essay, they provided a detailed account of the subjective relationships between each of the characters through a statement conveying a unique attitude for each of them. An individual is aware of his or her identity and orientation to the world by the manner in which they express their attitude.

3.Dictionary-based approach:

In dictionary-based methods, a collection of predefined opinion terms is assembled based on a set of criteria. In this technique, we assume that synonyms and antonyms have the opposite polarities to the parent word, but that synonyms have the same polarity to the parent word.

WordNet:

A huge lexical database in English can be found in WordNet [35]. In order to achieve this arrangement, nouns, verbs, adjectives, and adverbs are grouped into cognitive synonym sets (synsets), each of which represents a different conceptual idea. Across a synset, there are conceptual-semantic and linguistic connections that connect the synsets together. Due to the fact that WordNet allows users to categorize words according to their meaning, it appears that it is similar to a thesaurus. As described in [36], a word's polarity is determined by calculating the shortest distance between the words "good" and "bad." For our experiment, we extracted WordNet-containing terms from our lexicon to find out what could constitute a polarity.

General Inquirer:

An incredibly sophisticated text analysis tool, the General Inquirer (GI), is built upon one of the first hand-constructed lexicons ever produced. Since 1966, the GI has been developed and refined as an analysis tool to examine the content of documents. There are many aspects of communication that can be objectively identified by using a method like this, which is used by social scientists, political scientists, and psychologists [37]. At least 183 categories of words have been categorized in the lexicon, and there are approximately 11,000 words. The GI database contains a total of 1,915 positive words and 2,291 negative words. There have been a variety of studies conducted that have used it to assess the sentiment qualities of textual material.

LIWC:

The LIWC application is a tool that has been developed to study the many emotional, cognitive, structural, and process components that can be found in text samples of text. LIWC's proprietary vocabulary contains about 4,500 terms categorized into 76 groups, including about 905 terms in two areas critical to sentiment analysis that need to be categorized into one (or more) of the groups mentioned above. There are 406 words that describe positive emotions (for instance, Love, pleasant, good, fantastic). Conversely, there are 499 words that describe negative emotions (for example, Hurt, ugly, sad, awful, worse).

AFINN:

The AFINN [49] collection is comprised of English words covering a range of valence scores from -5 (negativity) to +5 (positivity). During the period between 2009 and 2011, Finn Rup Nielsen carefully labelled each of the words. There are a lot of tabs in this file. AFINN-96 is an original edition of the publication, containing 1468 unique words and phrases on 1480 lines, as opposed to AFINN-111 which contains 2477 words and phrases.

Deep learning

Deep learning algorithms are based on neural networks, and they perform better than traditional machine learning algorithms. However, a large quantity of data is required to train the model. As a result, they provide the highest level of accuracy in results when applied to massive datasets, as a result.

Some of the common deep learning methods are:

- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)
- Deep Belief Networks (DBN)
- Long-Short Term Memory (LSTM)

3.hybrid approach

As a hybrid strategy, machine learning is combined with lexicon-based strategies to generate outcomes. A hybrid sentiment analysis method refers to a combination of machine learning and lexicon-based approaches that are used for sentiment analysis. In the majority of systems that use hybrid technology, sentiment lexicons play an instrumental role in the integration of the two; in fact, hybrid technology mixes these two and is incredibly popular. Statistical and knowledge-based methods are used in sentiment analysis, which is a hybrid approach.

3.3 Techniques for sentiment analysis

In order to conduct a sentiment analysis project, there are several technologies that can be used including:

Natural language processing (NLP): NLP is a branch of artificial intelligence and computer science that focuses on how computers interact with human (natural) languages. In order to study and comprehend human language, machine learning and statistical models are used. To pre-process and comprehend text input, NLP is often utilized in sentiment analysis. The TF-IDF approach and the Bow method are two examples of NLP techniques that may be used to help with the process.

Machine learning: Machine learning methods can be utilized to train the models to categorize text as good, bad, or neutral depending on its content. Artificial neural networks (ANN), deep learning techniques like LSTM and bi-directional LSTM, Support Vector Machines (SVM),

Logistic Regression, Multinomial Naive Bayes, Random Forest, etc. are a few examples of machine learning algorithms and techniques. These are all instances of machine learning techniques.

Data visualization tools: Tools for data visualization may be used to clearly and attractively display the findings of a sentiment analysis research. Matplotlib, Seaborn, and Tableau are a few examples of data visualization software.

3.4 Software Tools Used for sentiment analysis:

A wide range of software applications, including, may be used to implement this system.

1. Google Colab
2. Jupiter Notebook
3. Anaconda

Google Colab:

Google Collaboratory, sometimes known as "Google Colab," is just an online free Jupyter environment that allows you to create and run code in a number of computer languages, such as Python, R, and TensorFlow. It offers access to a potent collection of computational resources, such as GPUs and TPUs, which can be used to train and execute massively scalable machine learning models. It is a fantastic tool for testing and experimenting with machine-learning models.

Google Drive is where Google Colab notebooks are kept and shared with other users. Due to easy access to the code and data used in a Colab notebook by other users, they are also a practical method to share and duplicate research.

In particular, Google Colab is helpful for machine learning academics and data scientists who wish to prototype and explore new model and concepts since it offers a simple environment for executing code and examining data. It is also a useful tool for learners who wish to put their newly acquired abilities to use in a practical situation, such as students and others.

Anaconda:

In order to make package organization and deployment easier, Anaconda is an open-source and free version of Python and R computer programming for scientific computing. It offers an easy approach to downloading and updating more than 1,500 open-source software.

Because it includes numerous well-known packages for information analysis and machine learning, including NumPy, Pandas, and scikit-learn, Anaconda is especially well-liked in the data analytics and machine learning communities. The virtual environment manager and conda package are also included, making it simple to establish and maintain distinct settings for different projects and to exchange those environments with others.

The Anaconda website allows for the download of Anaconda for Window panes, macOS, and Linux. The conda command-line program may be used to install more packages and build new environments once it has been installed.

3.5 Libraries for Sentiment Analysis:

As far as analysing the sentiments of a sentence or words is concerned, there are important libraries for Python that can be used in order to achieve that.

1. **NLTK (Natural Language Toolkit):** A Python module that offers instruments for dealing with human language data (text). It has tools for lemmatizing words, tokenizing text, and building n-grams.
2. **scikit-learn:** a Python library that offers a variety of machine learning methods, such as decision trees, support vector machines, and k-means clustering. Additionally, it has pre-processing tools for data, such as scaling, and normalization.
3. **Karas:** A Python package that enables the creation and training of deep learning models. For sentiment analysis, it may be used to build and train neural networks.
4. **Pandas:** a Python module that gives users the means to interact with data in the form of tables. It has tools for reading data in, cleaning it up so it's ready for analysis, and making visualizations.
5. **Matplotlib:** a Python module that enables you to build a variety of static, animated, and interactive visualizations. It may be used to display sentiment analysis project outcomes.
6. **Seaborn:** a Python package that offers Matplotlib's higher-level interface but is simpler to use for making specific visualizations, including heatmaps and time series plots.
7. **SpaCy:** a Python package with capabilities for natural language processing including dependency parsing, part-of-speech tagging, and tokenization. Working with a lot of text data requires a solution that may be quick and effective.
8. **Porter Stemmer:** A popular algorithm for reducing words to their stem, or fundamental form, is the Porter stemmer. By breaking down words to their most basic forms, the Porter stemmer may be used in sentiment analysis to determine the sentiment being represented in a piece of text. For instance, "running," "runs," and "ran" would all be shortened to the stem "run," which makes it simpler to understand the text's overall tone.
9. **Plotly:** Python has a module called Plotly that may be used to make interactive graphs and charts. Bar charts, line charts, scatter plots, and other types of charts may all be made with it. Plotly may be used in sentiment analysis to visualize the sentiment of a single text or a group of texts. For instance, you might use Plotly to make a bar chart displaying the proportion of favourable, unfavourable, and neutral feelings in a collection of tweets regarding a certain subject.
10. **Word Cloud:** Python's Word Cloud module allows you to make word clouds. A word cloud is a graphic depiction of the words that appear most often in a passage of text, with the size of each word according to how often it appears. The most frequent words and themes in a text or group of texts may be seen using Word Cloud in sentiment analysis. For instance, you might use Word Cloud to construct a word cloud of a collection of tweets about a certain subject and then analyse the word size to determine the tweets' most prevalent themes or feelings.
11. **TensorFlow:** Machine learning models, especially deep learning models, are often trained and deployed using it. Building and training machine learning models to categorize the sentiment of a piece of text may be done using TensorFlow in sentiment analysis. To construct and train a deep learning model that analyses a series of reviews

and forecasts whether each one is favourable or bad, for instance, you may use TensorFlow.

12. **Math:** A built-in Python library called "math" offers a variety of mathematical operations and functions. Numerous mathematical operations, such as simple arithmetic, trigonometry, statistical calculations, and others may be performed with it.

Chapter 4: Data Collection and Data Analysis

4.1 Data Collection

Data collecting is an important phase in a sentiment analysis study. In order to train a machine-learning model to categorize feelings as good, negative, or neutral, it is necessary to collect and label text data.

For a project including sentiment analysis, there are numerous methods for gathering data. Typical approaches comprise:

Using scrapers to access social media: Web scraping techniques may be used to gather information from websites such as Twitter, Facebook, and Instagram. Depending on the tone of the text, you may then categorize this data as positive, negative, or neutral.

collecting reviews: Reviews may be gathered from sites like Rotten Tomatoes and IMDb and classified as good or negative depending on how they are written.

Utilizing datasets that are accessible to the public: Numerous publicly accessible datasets with annotated text data are available and may be utilized for sentiment analysis.

The data you gather must be varied and accurate in terms of the emotion you are attempting to categorize. By doing this, the machine-learning model's accuracy will be enhanced.

Data for this project was gathered via the Kaggle website; a link is provided below.

<https://www.kaggle.com/datasets/krishbaisoya/tweets-sentiment-analysis>

About the data set:

Sentiment-140 is the name of the data collection. Utilizing the Twitter API, 1519832 tweets were retrieved and are included. The comments have been marked (0 = negatively, 1 = positively), and their emotion may be determined.

There are over 1.6 million tweets in the Sentiment140 dataset, each of which has the following characteristics:

- 1 Sentence: These are the tweets that the Twitter API was used to retrieve.
- 2 Sentiment: Text emotion, which may range from 0 (negative) to 1 (positive)

Below is a screenshot of what the dataset looks like. As a result, it illustrates how the phrases, and their feelings are expressed.

	sentence	sentiment
0	awww that s a bumner you shoulda got david car...	0
1	is upset that he can t update his facebook by ...	0
2	i dived many times for the ball managed to sav...	0
3	my whole body feels itchy and like its on fire	0
4	no it s not behaving at all i m mad why am i h...	0

Figure 1: Overview of the Dataset

4.2 Data Analysis:

A collection of feelings that have been classified as positive or negative make up the sentiment140 dataset. This is frequently used as benchmark dataset to assess the effectiveness of sentiment analysis algorithms.

We looked at the distribution of both positive and negative tweets as a starting point for our analysis of the sentiment140 dataset. To see this distribution, use a pie chart or a bar chart. The dataset's balance, with nearly equal amounts of good and negative tweets, or if it is biased in one way, will be fascinating to explore.

The most frequent terms used in tweets that were both favorable and negative are next. To show the most used terms, we may use a word cloud or a bar chart. It may be possible to get some understanding of the words that are most strongly linked to both good and negative emotion by doing this.

To categorize the sentiment of tweets in addition to these fundamental studies, you can consider developing a machine learning model. To train the model on the sentiment140 dataset and then evaluate its performance on another dataset, you may use a supervised learning approach like logistic regression or support vector machines (SVM).

In general, anybody involved in sentiment and natural language processing should use the Sentiment 140 dataset. We can better understand patterns and trends in social media user sentiment by examining this information.


```
[4] sns.countplot(x='sentiment', data=sentiment_df)
plt.title("Sentiment distribution")
```

```
Text(0.5, 1.0, 'Sentiment distribution')
```

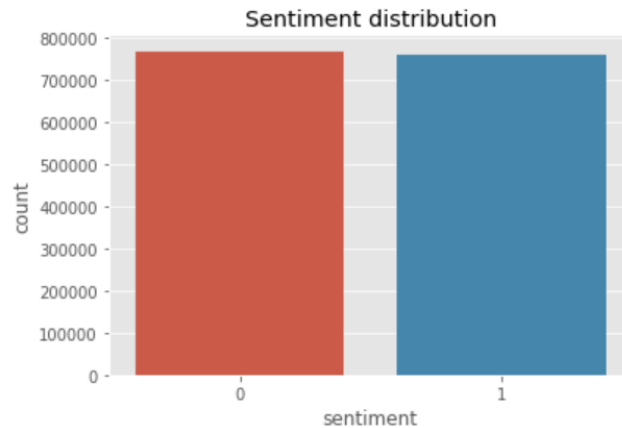


Figure 2: Visualize the Number of Positive and Negative Sentiment

Using Count Plot, the above Snapshot is used to provide a visual representation of the number of positive and negative reviews.

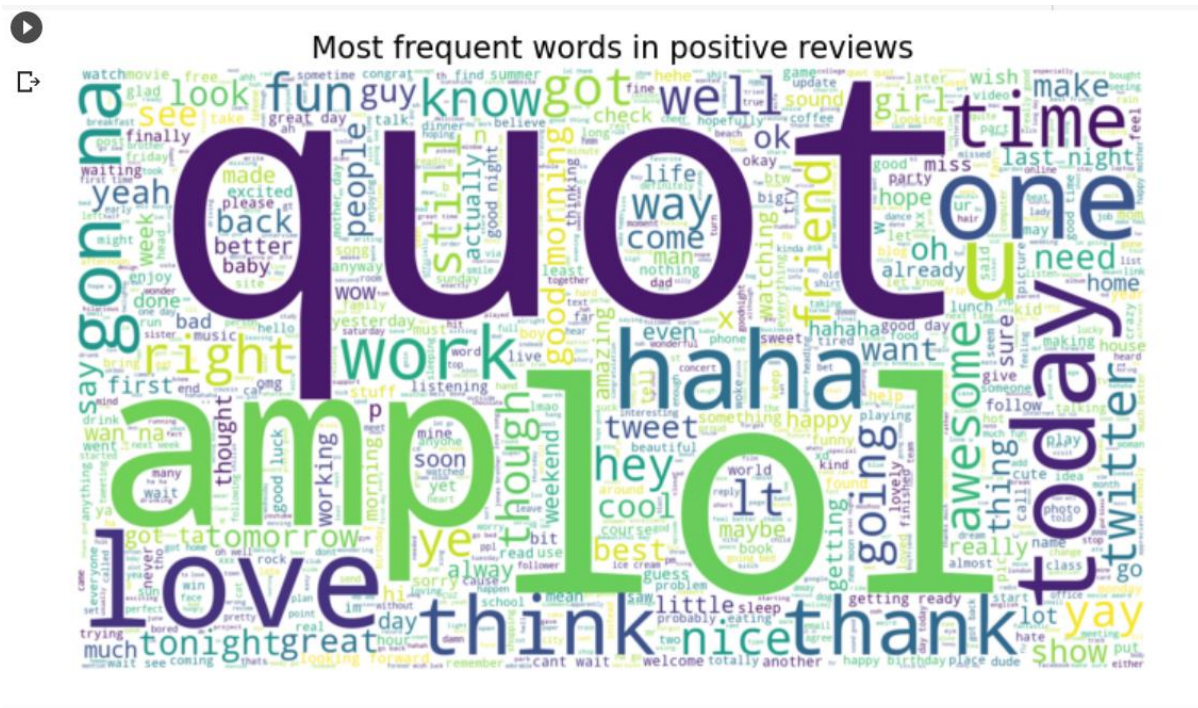


Figure 3: Visualize the Most Common Positive Review word

According to the Word Cloud Method described in the above Snapshot, the most common words in Positive Reviews are shown here.

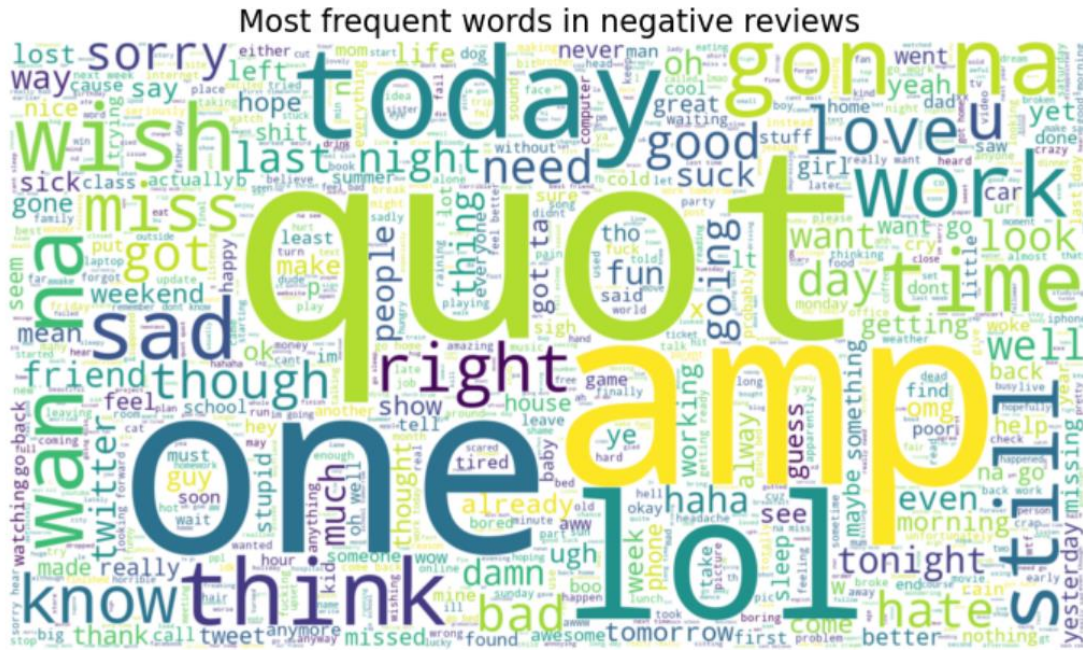


Figure 4: Visualize the Most Common Negative Review word

It can be seen in this snapshot that the most common words used in negative reviews are those that appear in a Word Cloud method described in the above snapshot.

	sentence	sentiment	word count
0	awww that s a bummer you shoulda got david car...	0	17
1	is upset that he can t update his facebook by ...	0	22
2	i dived many times for the ball managed to sav...	0	16
3	my whole body feels itchy and like its on fire	0	10
4	no it s not behaving at all i m mad why am i h...	0	23
...
1523970	just woke up having no school is the best feel...	1	11
1523971	thewdb com very cool to hear old walt interviews	1	9
1523972	are you ready for your mojo makeover ask me fo...	1	11
1523973	happy th birthday to my boo of alll time tupac...	1	12
1523974	happy charitytuesday	1	2

1523975 rows × 3 columns

Figure 5: Count the Number of words in Sentence

In order to count the number of words in a sentence, a snapshot like the one above can be used.

Chapter 5: Designing and Implementation

As part of the sentiment analysis process, there are a number of steps that must be followed:

1. Data Collection
2. Pre-processing of the data
3. Extracting features
4. Model of a train
5. Model used for the test
6. Evaluation of the model

5.1 Data Collection

For example, you can scrape data from social media platforms, online reviews, or other sources in order to gather this information.

Load the Data Set:

```
[ ] sentiment_df = pd.read_csv('/content/drive/MyDrive/Datasets/train_data.csv')
    sentiment_df.head()
```

	sentence	sentiment
0	awww that s a bummer you shoulda got david car...	0
1	is upset that he can t update his facebook by ...	0
2	i dived many times for the ball managed to sav...	0
3	my whole body feels itchy and like its on fire	0
4	no it s not behaving at all i m mad why am i h...	0

Figure 6: Load the data set

As you can see, this code will read the dataset from the URL that was provided and store it in a Pandas data frame. There are the following columns included in the dataset:

Sentence: In order to identify the sentiment, we must first identify the sentence

Sentiment: The label for the Dataset(0 = Negative, 1=Positive)

As a result of the head() function, we can print the data frame's first five rows.

Exploratory Data Analysis:

```

✓ [33] print(sentiment_df.info())
0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1523975 entries, 0 to 1523974
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  -
0   sentence    1523975 non-null  object
1   sentiment    1523975 non-null  int64
dtypes: int64(1), object(1)
memory usage: 23.3+ MB
None

```

Figure 7: Summarization of the Data Frame

The picture above showed A Data Frame may be quickly summarized using the info () method. The non-null values, memory utilization, index and column types, and other details about a Data Frame are printed by this function.

```

✓ [34] print(sentiment_df.sentiment.value_counts())
0s

0    767059
1    756916
Name: sentiment, dtype: int64

```

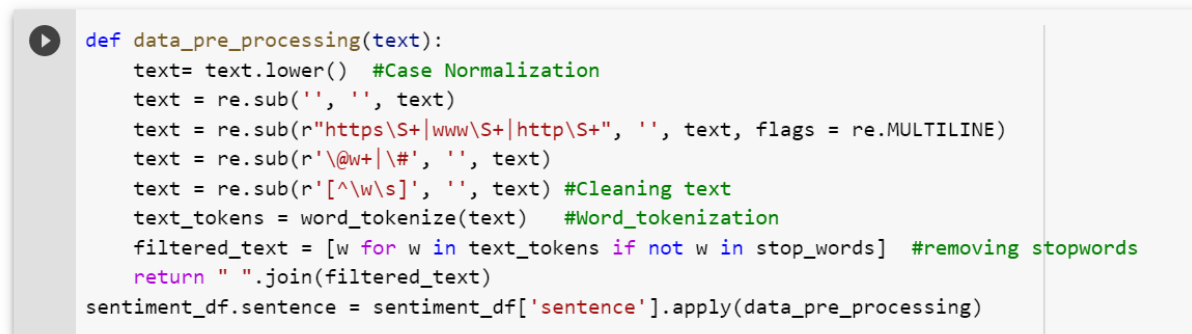
Figure 8: Counts of the distinct Value

The snapshot seen above showed The object holding counts of distinct values is returned by the value counts() method.

5.2 Preprocessing of the data:

The pre-processing step of the data workflow consists of cleaning up noisy, incomplete, and inconsistent data during the data cleaning process. This is considered to be a pre-processing process because it refers to the process of managing data that has not been evaluated yet. This is in order to identify characteristics that might be significant. Prior to using the data in the

feature selection job for the purpose of selecting features, it is necessary to process the data beforehand. The following are some of the tasks involved in pre-processing:



```
def data_pre_processing(text):
    text= text.lower() #Case Normalization
    text = re.sub(',', '', text)
    text = re.sub(r"https\S+|www\S+|http\S+", '', text, flags = re.MULTILINE)
    text = re.sub(r'\@w+|\#', '', text)
    text = re.sub(r'^\w\s', '', text) #Cleaning text
    text_tokens = word_tokenize(text) #Word_tokenization
    filtered_text = [w for w in text_tokens if not w in stop_words] #removing stopwords
    return " ".join(filtered_text)
sentiment_df.sentence = sentiment_df['sentence'].apply(data_pre_processing)
```

Figure 9: Apply Data Pre-processing

The above image shows the process of applying data pre-processing. The steps for this process can be found below.

Start by learning how to perform steps such as the following when pre-processing text using NLTK:-

- The URL, special characters, numbers, punctuation, and other elements should be removed.
- It is necessary to remove stop words from the sentence.
- The removal of Retweets (in the case of the Twitter dataset)
- The stemming processes
- Tokenization
- Lemmatization
- POS Tagging

Tokenization

Tokenization is one of the first steps in the process of doing text analytics. In the process of tokenization, larger segments of text, such as sentences or words, are broken up into smaller chunks, such as individual words or phrases. This is designed to be easy to use and understand. Tokens, which act as building blocks for the construction of a phrase or paragraph, are separate entities that form the basis of the construct. This method can be used to quickly and easily get rid of tokens that are no longer required by using them. During the tokenization process, individual words and phrases are extracted from raw text so that they can be converted into tokens. The use of these tokens can both serve as a tool for understanding the context of the conversation or for the construction of a model for natural language processing. This model can be used for both purposes. Tokenization, which is a software program that analyses the order in which words are arranged in the text, can aid readers in understanding the meaning of a text a bit better.

Sentence Tokenization

Using the sentence tokenizer, the paragraphs of text are separated into individual sentences by the tokenizer.

Word Tokenization

Word tokenization refers to the act of separating a vast amount of text into its component words one at a time. The tokenization of words is an essential step in the process of sentiment analysis in order to assess sentiment. Because of this, we are able to examine the sentiment of individual words and phrases, rather than just the overall sentiment of the entire text. This is because it is possible to analyse the sentiment of individual words and phrases.

Stop words

In many languages around the world, there are words that do not have any meaning whatsoever. Our goal in deleting these terms is to remove low-level information from our text as it allows us to focus on the key information, which in this case is articles, by eliminating low-level information from our text. There are certain words that are very common in the English language, such as "if," "but," "we," "he," "she," and "them," and "stop words." It is quite often possible to remove certain terms from a text with minimal or no impact on its meaning, and doing so often (although not always) improves the performance of the model as a result.

Removing Stop words

In every human language, there are several words that don't mean the same thing in different contexts. The aim of removing these terms from our text is to remove low-level information from it, thereby enabling us to concentrate more on key information, such as articles, rather than the terms that we are currently using.

POS Tagging

POS tagging is a technique for detecting grammatical categories of words by identifying the part of speech (POS) that the given words belong to. Depending on the context, the word will be classified as a noun, pronoun, adjective, verb, or adverb. POS Tagging is the process of adding a tag to a word based on its relationship with other words in the phrase that it appears inside.

Case Normalization:

The process of normalising a word, statement, or document is performed based on its case. In other words, when the whole word, statement, or document is normalised based on its case, it is transformed into lower case or upper case.

Lexicon Normalization¶

There is another kind of textual noise that is taken into account during the lexicon standardization process. There are many derivatives of the word connection, such as connected, connected, connecting, which can all be condensed into the single word "connect." In order to achieve this, we will reduce all of the derivatives of the word to the base term that represents all of them.

```
[45] #Lexicon Normalization
      #performing stemming and Lemmatization
      w_tokenizer = nltk.tokenize.WhitespaceTokenizer()
      lemmatizer = nltk.stem.WordNetLemmatizer()
      def lemmatize(data):
          filter_text = [lemmatizer.lemmatize(word) for word in w_tokenizer.tokenize(data)]
          return " ".join(filter_text)
```

Figure 10: Performing Stemming and Lemmatization

Stemming

The process of stemming, which can be explained as the reduction of words to their word roots or the removal of their semantic affixes, is a form of linguistic normalization. A stemming process involves removing the suffix from a word and reducing it to the term it stands for under the suffix. The most basic form of a token is to boil it down to its most basic components. As an example, if you combine the words connection, connected, and connecting, you get the word "connect".

Lemmatization

In lemmatization, words are reduced to their linguistically correct lemmas, or root words, according to their level of development. By analysing the lexical and morphological meaning of the root word, the word is altered. A lemmatization process is generally more complex than a stemming process. Stemmers are programs that analyse a single word without taking into account its context. For instance, the word "better" has a lemma of "good." In this case, stemming will be failed since a dictionary check will be required to determine the correct stem.

5.3 Extracting Feature:

Feature extraction refers to the process of selecting and extracting from a dataset the characteristics that are most relevant to and informative about a subject (for the purpose of feeding those characteristics into a machine learning model) and then processing them for the purpose of assimilating them into a model. If you apply the context of doing sentiment analysis on the Sentiment140 dataset to the context of reviewing the tweets, you will need to determine which words and phrases in the tweets indicate positive or negative sentiment based on their use in the tweets.

The techniques available for extracting features from a dataset depend on the features of the dataset as well as your study's objectives. In addition, they depend on, they depend on what type of features you need. Among the most common approaches are:

```
[59] def make_encode_vector():
    X = sentiment_df['sentence']
    Y = sentiment_df['sentiment']
    vector = TfidfVectorizer()
    le = LabelEncoder()
    X = vector.fit_transform(sentiment_df['sentence'])
    Y = le.fit_transform(sentiment_df['sentiment'].values)
    x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=42)
    return x_train, x_test, y_train, y_test
x_train, x_test, y_train, y_test= make_encode_vector()
```

Figure 11: Performing Feature Extraction

Bag-of-Words:

Models are simple approaches that create feature vectors for each document (or tweet in this case), which include elements that represent a specific word or phrase and their values indicating whether or not they are present in the text. This method generates feature vectors for each document in a straightforward manner. A bag of words is analysed as a whole, regardless of the order in which the words appear.

Term frequency-inverse document frequency (TF-IDF):

This method is similar to the bag-of-words model in that it considers the overall frequency of the term throughout the whole dataset as well. In the feature vector, words with a high frequency inside a single text but a low frequency overall will be given a higher weight.

Word embeddings:

instead of expressing each word as a single feature, this technique uses dense vectors (known as "embeddings") in high-dimensional spaces. Word embeddings are an example of a technique that captures the meaning of a word as well as its context, and they can be used in machine learning models as features. This method is called word embedding training and can be taught on large datasets or practiced on external datasets before being fine-tuned for a specific task.

5.4 Feature Selection:

Sentiment analysis is highly challenging because the most difficult part is selecting relevant features from the pre-processed text in order to carry out sentiment analysis. In order to reduce the size of the feature space as well as the computing expenses associated with the selection of features, the main objective of the feature selection process is to reduce the size of the feature space. A feature selection process can mitigate the overfitting of a learning scheme with regard to training data by allowing the scheme to be adapted to the data. Often, in order to choose features, point-of-sale (POS) labels are used in conjunction with feature selection strategies.

Among the probable characteristics that are likely to be considered in an analysis of the sentiment included in the sentiment140 dataset, there are the following that might be considered as well:

There are three common types of features that are used in machine learning algorithms, including unigrams, bigrams, and ngrams. It was found that these methods worked best when they were applied to a sentiment evaluation dataset and analysed using various strategies for choosing features.

1. The use of words that are strongly associated with positive or negative emotions, such as the words "love" or "hate," should be avoided.
2. You may convey the mood of a tweet through the use of emoticons, which, for example, include expressions of happiness and sadness.
3. If words are written entirely in capital letters, for instance, it's possible that the sense of power conveyed by those words would be more powerful.
4. Tweets with the most information and context tend to be those that are longer, and are therefore more likely to provide more information and context.
5. There may be hashtags or mentions present in the tweet, which may serve as more context or indicate who is being addressed in the tweet
6. The words in the tweet should be tagged with part of speech tags, given that certain parts of speech can enhance the impression that the tweet is conveying a particular mood (for example: adjectives).
7. Wordlists with a sense of sentiment are the kinds of lexicons that categorize words according to whether they have a positive or negative meaning associated with them
8. A N-gram is a continuous word sequence from beginning to end and is often called a word sequence from beginning to end. For instance, the pairs of phrases "love you" and "so happy" are both examples of bi-grams, which have an N-count of 2.
9. The use of syntactic parse trees in order to provide information about the structure of a phrase, as well as a means to provide insight into the relationships between individual terms within the phrase, can prove to be a very useful tool.
10. There are various types of embeddings of words that are computed using extensive datasets of words and are also known as numerical representations of words. As a function of capturing semantic links between words, word embeddings can prove useful for sentiment analysis, due to their ability to capture semantic links between words.

5.5 Training and testing the model

Model training: A machine learning model will be trained to categorize text as positive, negative, or neutral using the retrieved attributes. Support vector machines, decision trees, and deep learning models are just a few of the methods that may be used to this.

Test model: After the model has been trained, must assess its performance using a test set of data. then assess the model's classification accuracy on brand-new, unread sentiment.

Divided test and training set:

Separating the dataset into a training set and a test set is a smart technique to determine model performance.

Using the function train test split, divide the dataset (). essentially 3 factors that must be passed: features, target, and test set size. Additionally, choose records at random using random state. As shown in the image below.

```
[ ] from sklearn.model_selection import train_test_split
    X = sentiment_df['sentence']
    Y = sentiment_df['sentiment']
    x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.3, random_state=42)
```

Figure 12: Split the Data into Training and Testing

The following section discusses some of the different machine-learning algorithms that are used.

1. Logistic Regression:

When dealing with huge text data sets, the approach of logistic regression is often utilized in sentiment analysis projects. It's a statistical technique for forecasting binary outcomes, like positive or negative emotion.

```

▶ #Logistic regression
def logistic_reg_model(x_train, x_test, y_train, y_test):

    # Create a logistic regression mode
    logistic_reg = LogisticRegression()

    # Train the model on the training data
    log_reg_model = logistic_reg.fit(x_train, y_train)

    # Make predictions on the test data
    logistic_pred = logistic_reg.predict(x_test)

    # Calculate the model's accuracy
    logistic_acc = accuracy_score(y_test, logistic_pred)

    return log_reg_model,logistic_pred,logistic_acc,y_test
log_reg_model,logistic_pred,logistic_acc,y_test = logistic_reg_model(x_train, x_test, y_train, y_test)

```

Figure 13: Logistic Regression Model

Import the logistic regression class from the Sklearn library into this code first. build a Logistic Regression class object, and then assign it to the model variable. The model is then trained using the fit technique using training data given in the form of the X train and y train variables.

By invoking the prediction function and sending it the X test data once the model has been trained, we can use it to make predictions on the test data. The logistic pred variable contains the outcomes' predictions. Finally, we determine the model's correctness on the test data using the score approach. This is accomplished by contrasting the real labels of the test data (stored in y test) with the model's predictions (stored in logistic pred).

2. Multinomial Naïve Bayes:

A classification technique called Multinomial Naïve Bayes is often used for sentiment analysis and other natural language processing applications. It is built on the concept of leveraging the likelihood that each word will appear in a certain class (such as positive or negative emotion, for example) to create predictions about the class of new, unforeseen data.

An example of Python code that uses Multinomial Naïve Bayes is shown here:

```

▶ # Model Generation Using Multinomial Naïve Bayes
def naive_bayes_model(x_train, x_test, y_train, y_test):

    # Create a Multinomial Naïve Bayes model
    mnb_model=MultinomialNB()

    # Train the model on the training data
    mnb_model.fit(x_train, y_train)

    # Make predictions on the test data
    mnb_model_pred = mnb_model.predict(x_test)

    # Calculate the model's accuracy
    mnb_model_acc = accuracy_score(y_test,mnb_model_pred)

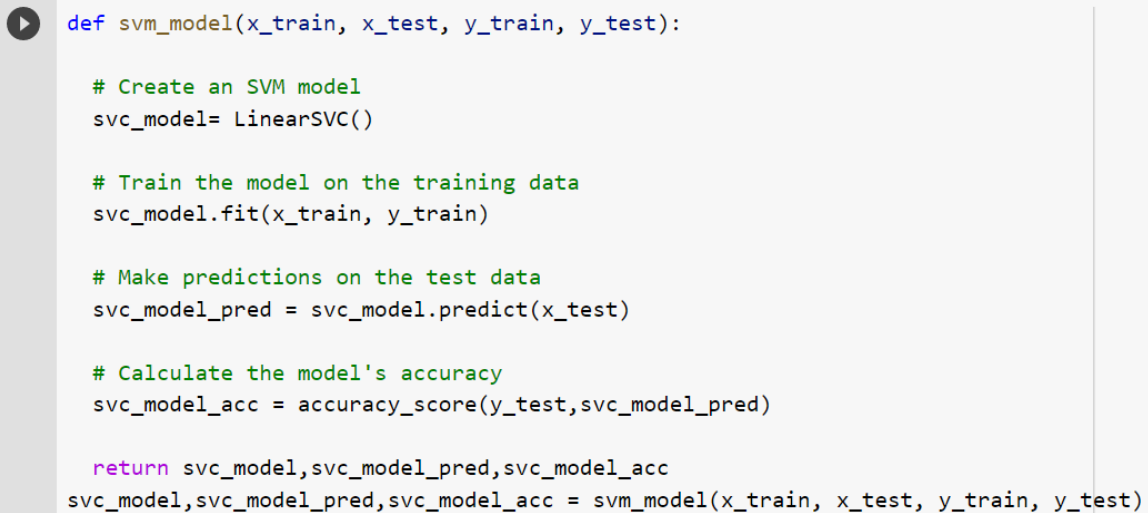
    return mnb_model,mnb_model_pred,mnb_model_acc
mnb_model,mnb_model_pred,mnb_model_acc = naive_bayes_model(x_train, x_test, y_train, y_test)

```

Figure 14: Multinomial Naïve Bayes Model

3. Support vector machines (SVMs):

There is a necessity to point out that these are robust models that can be used to solve classification problems, such as sentiment identification, by applying them. In order to ensure that they are performing properly, all the classes are divided into hyperplanes that provide the most differentiation among them.



```
def svm_model(x_train, x_test, y_train, y_test):

    # Create an SVM model
    svc_model= LinearSVC()

    # Train the model on the training data
    svc_model.fit(x_train, y_train)

    # Make predictions on the test data
    svc_model_pred = svc_model.predict(x_test)

    # Calculate the model's accuracy
    svc_model_acc = accuracy_score(y_test,svc_model_pred)

    return svc_model,svc_model_pred,svc_model_acc
svc_model,svc_model_pred,svc_model_acc = svm_model(x_train, x_test, y_train, y_test)
```

Figure 15: Support Vector Machine Model

The features and labels for the training data in this code are `x_train` and `y_train`, respectively, whereas the features and labels for the test data are `x_test` and `y_test`. The `predict` approach is used to make predictions on the test data, whereas the `fit` method is used to train the model on the training data. The accuracy of the model, which is the percentage of accurate predictions based on the test data, is determined using the `score` technique.

The Multinomial Model generates predictions based on the presumption that the characteristics are conditionally independent given the class and that a multinomial distribution determines the likelihood of each feature belonging to a certain class. Making predictions based on the class with the greatest probability, it then calculates the class probabilities using the maximum likelihood estimate.

4. LSTM

For applications involving natural language processing, such as sentiment analysis, the recurrent neural network (RNN) known as Long Short-Term Memory (LSTM) is often used. A particular kind of neural network called an RNN is designed to handle sequential data, such as text or time series data. They do this by using hidden states, which have the ability to recall data from earlier time steps and so enable them to generate predictions based on long-term relationships in the data.

```
[66] # Define the model architecture
model = Sequential()
model.add(Embedding(500, 120, input_length = X.shape[1]))
model.add(SpatialDropout1D(0.4))
model.add(LSTM(176, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(2,activation='softmax'))

# Compile the model
model.compile(loss = 'categorical_crossentropy', optimizer='adam', metrics = ['accuracy'])
print(model.summary())
```

WARNING:tensorflow:Layer lstm will not use cuDNN kernels since it doesn't meet the criteria.
Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 50, 120)	60000
spatial_dropout1d (SpatialD ropout1D)	(None, 50, 120)	0
lstm (LSTM)	(None, 176)	209088
dense (Dense)	(None, 2)	354

=====
Total params: 269,442
Trainable params: 269,442
Non-trainable params: 0

Figure 16: Long Short Term Memory Model

The LSTM, Sequential, Dense, and Embedding layers are initially imported into this code using the keras library. The next step is to design the model architecture by building a sequential model with an embedding layer, an LSTM layer, and a dense output layer with a sigmoid activation function.

Following that, we construct the model by defining the loss function (binary cross entropy) and the optimization procedure (Adam). Then, using a batch size of 32, we train the model using the fit technique on the training phase for 10 iterations. The model is finally evaluated using the test data by using the evaluation method and handing it the X test and y test inputs. Loss and accuracy values as a consequence are kept in the corresponding variables.

Chapter 6: Model Performance Evaluation Metrics

To assess the effectiveness of a sentiment analysis model, a variety of assessment measures may be used. Below are a few such examples:

Accuracy: A simple way to measure how well the model predicts the future is to measure how many of its predictions are correct, which is the most basic and straightforward metric.

Precision: This gauges the percentage of optimistic forecasts that come true. A model's accuracy, for instance, may be expressed as a number, such as 0.8, which indicates that, of all the positive predictions it made, 80% of them came true.

Recall: This indicator shows what percentage of real positive instances the model properly anticipated. A recall of 0.7, for instance, indicates that the model accurately predicted 70% of the actual positive occurrences.

F1 score: It is often used as a single statistic to assess the overall effectiveness of a model. This is the harmonic means of accuracy and recall.

AUC-ROC curve: This graphical figure displays how well a binary classifier performs. On the y-axis, it displays the genuine positive rate (recall), while on the x-axis, it displays the false positive rate.

Confusion matrix: This table provides a summary of a classification model's performance by displaying the proportion of true positives, false positives, true negatives, and false negatives.

Classification report: Including accuracy, recall, f1-score, and support, this table summarizes a classifier's performance (the number of samples in each class).

It is challenging to predict which model would perform best in a certain sentiment analysis job since accuracy will vary depending on a number of variables, such as the amount and quality of training data, the difficulty of the task, and the model hyperparameters used.

Support vector machines (SVMs), which may both attain high accuracy, are often utilized for sentiment analysis jobs in general.

Another well-liked option for sentiment analysis, particularly when dealing with text data, is Multinomial Naive Bayes (NB). This is so because the multinomial NB model is reasonably quick to train and can handle a lot of features, such the quantity of words in a text. While this may not always be the case in sentiment analysis tasks, it makes the assumption that the characteristics are independent given the class.

Recurrent neural network (RNN) models with long short-term memory (LSTM) are especially effective at processing sequential input, including text. They have been successfully used to a number of NLP tasks, including sentiment analysis. To obtain high performance, LSTMs may

need more data and be more computationally costly to train than other models.

The appropriate strategy will ultimately rely on the details of the data and the job at hand. In order to choose the optimal model for a particular job, it is often essential to test out many models and assess their performance on a validation set.

Logistic Regression:

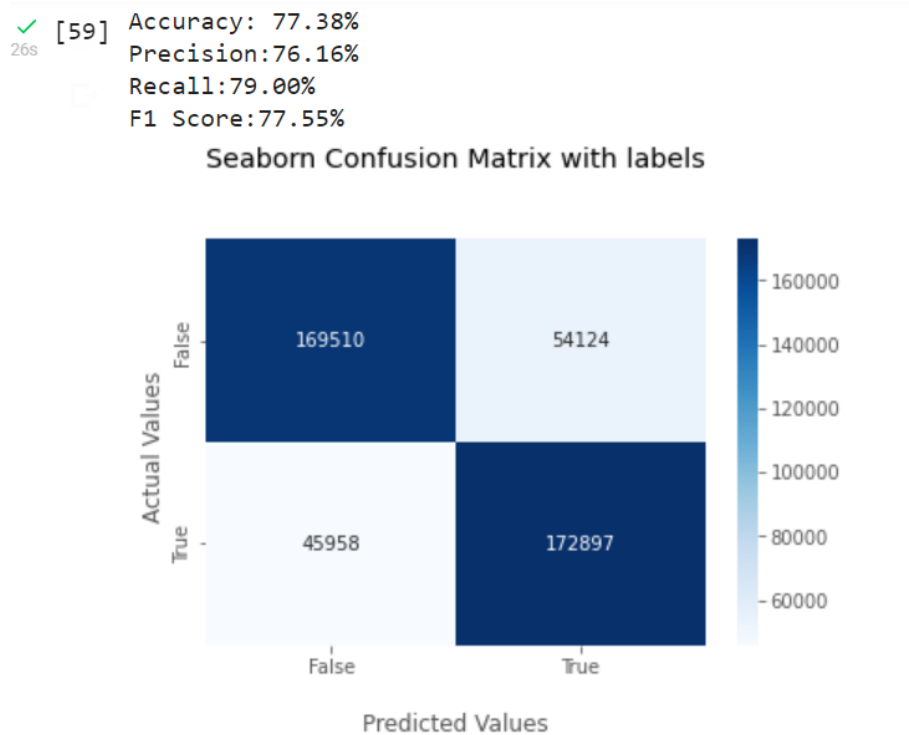


Figure 17: Confusion Matrix for Logistic Regression

Multinomial Naive Bayes:

Accuracy: 75.57%
 Precision:76.09%
 Recall:73.80%
 F1 Score:74.93%

Seaborn Confusion Matrix with labels

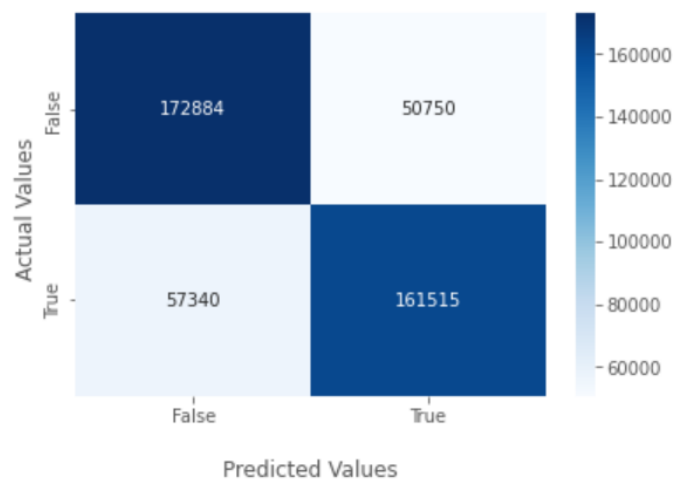


Figure 18: Confusion Matrix for Multinomial Naïve Bayes

Support vector machines:

Accuracy: 76.68%
 Precision: 75.54%
 Recall: 78.18%
 F1 Score: 76.83%

Seaborn Confusion Matrix with labels

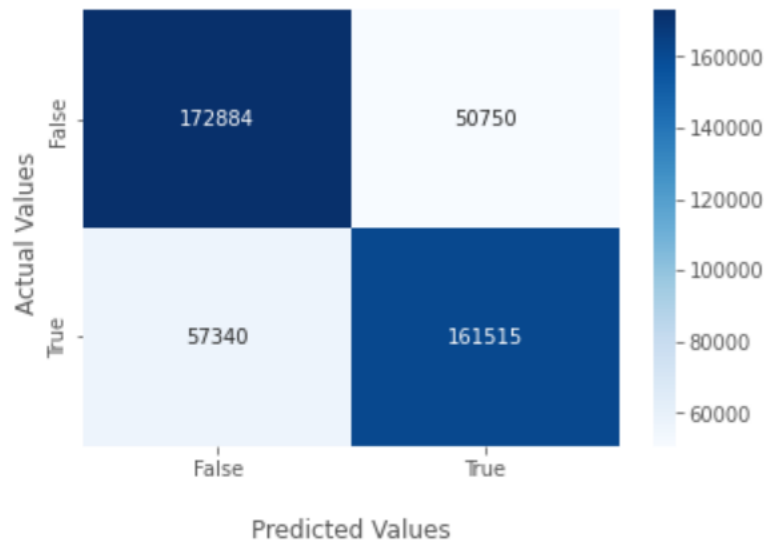


Figure 19: Confusion Matrix for Support Vector Machine

Real-world Application of sentiment analysis:

Sentiment analysis in social media:

There are a number of social networking websites, such as Twitter and Facebook, among others. In these websites, individuals are able to express their thoughts and ideas on a variety of issues today. One of the most common places for them to do so is on Twitter and Facebook. In order to be able to take advantage of this information for the purpose of marketing and expanding a company, it is important for marketers to understand what consumers think about a product or service before committing to marketing campaigns or programs. A lot of tweets of a particular good or service are associated with a general tone through sentiment analysis, which is applied to a large number of tweets that are of a certain tone associated with the product or service. Marketing professionals can then use this information to inform their deliberations regarding the approach they should take with their upcoming marketing campaigns.

The sentiment analysis of customer reviews:

There are a number of social networking websites, such as Twitter and Facebook, among others. In these websites, individuals are able to express their thoughts and ideas on a variety

of issues today. One of the most familiar places for them to do so is on Twitter and Facebook. In order to be able to take advantage of this information for the purpose of marketing and expanding a company, it is imperative for marketers to understand what consumers think about a product or service before committing to marketing campaigns or programs. Many tweets about a particular good or service are associated with a general tone through sentiment analysis. This is applied to a large number of tweets that are of a certain tone associated with the product or service. Marketing professionals can then use this information to inform their deliberations regarding the approach they should take to their upcoming marketing campaigns.

Sentiment analysis for stock market prediction:

Machine learning models can be used to predict whether the stock prices of various firms will increase or decrease over a certain time period. This is because the stock prices of various firms are subject to a great deal of volatility. On social media sites, such as Reddit, Twitter, and others, there is often a discussion about certain stocks which can be read.

Chapter 7: Conclusion

In conclusion, the Sentiment140 dataset is an important tool for sentiment analysis research. It has been extensively utilized in several research investigations and comprises a sizable collection of data that have been manually categorized as either positive or negative. Even though the dataset is relatively limited in scope since it only contains phrases with good or negative emotions, it is nevertheless an effective method for figuring out and forecasting the sentiment of social media postings. In conclusion, the Sentiment140 dataset is a valuable tool for scholars interested in sentiment analysis and is expected to be heavily used in the future.

Long short-term memory (LSTM) networks, logistic regression, multinomial naive Bayes, support vector machines, and other models have all been tested for their effectiveness in sentiment analysis.

Overall, we discovered that all four models performed well on the sentiment analysis test, with the logistic regression model achieving the best accuracy. But the LSTM model also needed the greatest computing power and the most time to train.

The Multinomial NB model was the quickest to train while still achieving excellent accuracy, however, the LSTM and SVM models also performed well and were quite quick to train.

In a further study, it would be intriguing to investigate other model types, such as transformer-based models, which have recently attained cutting-edge outcomes on a range of natural language processing tasks.

This research emphasizes the value of carefully comparing many models to determine which Logistic Regression is best suited for the job at hand.

Chapter 8: Future work

The Sentiment140 dataset is a crucial resource for sentiment analysis research, to sum up. It contains a vast collection of data that have been manually classified as either positive or negative and has been widely used in several research endeavours. Even though the dataset is rather limited in scope since it only includes words associated with positive or negative emotions, it is nevertheless a useful technique for determining and predicting the mood of social media posts. The Sentiment140 dataset is a useful resource for academics interested in sentiment analysis, and it is anticipated that it will be actively utilized in the future.

Support vector machines, logistic regression, multinomial naive Bayes, long short-term memory (LSTM) networks, and other models have all been evaluated for their efficacy in sentiment analysis.

Overall, we found that the sentiment analysis test showed good performance from all four models, with the logistic regression model attaining the greatest accuracy. However, the LSTM model also required the most training time and processing resources.

The LSTM and SVM models both performed well and trained pretty quickly, but the Multinomial NB model was the fastest to train while still obtaining good accuracy.

It would be fascinating to look at additional model types in subsequent research, such as transformer-based models, which have recently achieved cutting-edge results on a variety of natural language processing tasks.

The need to carefully examine many models to decide which Logistic Regression is most appropriate for the task at hand is emphasized by this study.

Reference

1. Nasukawa, T. and Yi, J. (2003) *Sentiment analysis: Capturing favourability using natural language processing*, pp. 70.
2. Hussein, Doaa Mohey El-Din Mohamed (2018) 'A survey on sentiment analysis challenges', *Journal of King Saud University - Engineering Sciences*, 30(4), pp.330-338. doi: <https://doi.org/10.1016/j.jksues.2016.04.002>.
3. Jagdale, R.S., Shirsat, V.S. and Deshmukh, S.N. (2016) 'Sentiment analysis of events from Twitter using open source tool', *International Journal of Computer Science and Mobile Computing*, 5(4), pp.475-485.
4. Hu, M. and Liu, B. (2004) *Mining opinion features in customer reviews*, pp. 755.
5. Hu, M. and Liu, B. (2004) *Mining and summarizing customer reviews*, pp. 168.
6. Bollen, J., Mao, H. and Zeng, X. (2011) 'Twitter mood predicts the stock market', *Journal of*

- Computational Science*, 2(1), pp.1-8.
7. Efron, M. (2004) *Cultural Orientation: Classifying Subjective Documents by Cociation Analysis*. pp. 41.
 8. J. Ramteke *et al.* (2016) *Election result prediction using Twitter sentiment analysis*, pp. 1.
 9. Berg, O. (1975) 'Health and Quality of Life', *Acta Sociologica*, 18(1), pp.3-22. doi: 10.1177/000169937501800102.
 10. Pang, B., Lee, L. and Vaithyanathan, S. (2002) 'Thumbs up? Sentiment classification using machine learning techniques', *arXiv Preprint Cs/0205070*,
 11. Turney, P.D. (2002) 'Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews', *arXiv Preprint Cs/0212032*,
 12. Hatzivassiloglou, V. and McKeown, K. (1997) *Predicting the semantic orientation of adjectives*, pp. 174.
 13. Tong, R.M. (2001) *An operational system for detecting and tracking opinions in on-line discussion*, .
 14. Wiebe, J. (2000) 'Learning subjective adjectives from corpora', *Aaai/Iaai*, 20(0), pp.0.
 15. Pang, B. and Lee, L. (2005) 'Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales', *arXiv Preprint Cs/0506075*,
 16. Hu, M. and Liu, B. (2004) *Mining and summarizing customer reviews*, pp. 168.