University of Essex
Department of Mathematical Sciences

MA981: DISSERTATION

Sentiment Analysis on IMDB movie review dataset using Machine Learning, Neural Network and Transformer

Atharva Milind Rodge
Registration number: 2311527

Supervisor: Lan Truong

September 17, 2024
Colchester

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Context

Sentiment Analysis or emotion detection, a crucial part in this technological advancement and a part of Natural Language Processing (NLP). Sentiment analysis is a step which involves identifying sentiment of a piece of text or a sentence for our analysis. This process determines three types of sentiments positive, negative, or neutral for a given piece of text. it's often use in various businesses to understand customer behaviour, for example: on e-commerce site, if a customer reviews a products and gives it one star to that particular product then the other customers they are unlikely to buy that product. Moreover, if a stock price tumbles because of a negative news about the company, that can be identified by analysing the sentiment and using historic data. Analysing sentiment has an invaluable advantage for the companies and businesses to understand customer behaviour and opinion, this makes it easier for the businesses to take decisions. [33]

The main objective of the project is to transform the qualitative data into useful insights to make informed decisions based on the classification of the sentence/text given by general public. For example, if we consider a clothing brand which has a huge customer base and they are getting reviewed bad for one of their latest product, the company can take immediate action based on the sentiment of the reviews and stop selling such kind of products to maintain their value in the market.[33]

Analysing sentiment also plays crucial role in market research, social media, stock predic-

tions, trend analysis, customer behaviour, user experience. By using sentiment analysis one can improve their offers and work on better outcomes. [33]

1.2 Problem statement

Sentiment analysis is a critical task in natural language processing (NLP). It is also one of the important task in many industries and businesses like marketing, stock predictions, social media, movie's review, tweet analysis, e-commerce and customer behaviour. Despite the era of 'Gen-Ai' and 'LLM' analysing sentiment still face many challenges and slows down the accuracy of the analysis.

One major challenge is the complex and large amount of datasets with uncertain and complex human language. Cleaning this data, removing stop-words and identifying and removing neutral words which won't affect the sentiment of the piece of text is a real task. Additionally, the text might also content some links, emojis, number and special characters which will eventually impact the accuracy of the sentiment, so it is important to process and clean the text before working on it. [30]

Traditional machine learning languages struggle in predicting the sentiment classification or emotion, it limits itself to specific domains but still can be worked on for predicting sentiment. Neural network like artificial neural network (ANN), convolutional neural network (CNN) etc have advantages when it comes to NLP, sentiment analysis or working with textual data. These techniques naturally learn from their features without manual feature engineering, but struggle with long texts and are hard to train. [30]

Transformers like BERT, RoBERTa and GPT has advantages over these drawbacks. Transformer can work on large textual data. however, Transformers takes time to get trained depending on the batch size, data size and number of iterations. The Field of LLM, Gen-AI are the future of textual data and predicting sentiment with accuracy will be a important aspect of this. [30]

1.3 Objective and aim of the project

The objective of the study is to take a deep dive into the methodologies and techniques of predicting the sentiment of the text that is sentiment analysis. Training large amount of data

to predict the sentiment of the text in our dataset. We have choose to work on IMDB movie's review dataset. Additionally we will also use Natural language processing (NLP) for data prepossessing like removing stop-words, lemmatization etc. We will also discover some applications and techniques such as standard machine learning algorithms, neural network and deep learning based transformer approach to identify the sentiment in different types of text. Thereafter, we will compare the results of standard machine learning algorithms, neural networks and BERT transformer.

1.4 Significance of the Study

This project will contribute to the field of natural language processing (NLP) by understanding key challenges in sentiment analysis by using various techniques like Machine learning, Neural Network like ANN and transformers like RoBERT on IMDB review dataset offering improved and practical benefits for industries to understand textual data and depict meaning from it.

Predicting sentiment analysis is important for businesses to understand textual data to depict meaningful insights and customer behaviour, guided product development and unique marketing strategies. Improved sentiment analysis models provide deeper insights to meet consumer needs. In the film industry, analysing movie reviews helps studios, marketers and film makers understand how audiences feel about the films. Depending on the industry this helps target consumers and customers effectively. Better the sentiment analysis tool better it will help to manage a company's reputation by identifying potential problems. [31]

By improving sentiment analysis techniques by implementing machine learning (ML), neural network (NN) and transformers. We will aim to enhance our accuracy. This study is useful for businesses and industries improving our knowledge of sentiment analysis in natural language processing (NLP). [31]

1.5 Scope of study

- The scope of this study is on sentiment analysis using machine learning (ML), neural network and transformers.
- In this study we are working on IMDB review dataset. The large dataset contains

collection of movie reviews classified as positive and negative.

- This study focuses on comparing different techniques like machine learning methods which includes SVM, Random forest, etc. Neural Network method like ANN and RoBERTa transformer.
- By comparing these models we aim to identify the best technique with highest accuracy and strengths and weakness between these techniques for analysing sentiment.
- We will measure performance of these techniques using metrics like accuracy, precision, recall, and F! score. This evaluation will help us understand the reliability of each model.

1.6 My Contribution

In this study, we applied machine learning, neural networks, and BERT transformer techniques to predict the sentiment of movie reviews. We compared traditional machine learning models with neural networks and BERT transformers to identify the best method for predicting sentiment from text. Additionally, we focused on optimizing the models, particularly aiming to reduce the training time for the BERT transformer model.

We implemented each model, analyzed the results by plotting performance curves and confusion matrices, and also plotted ROC curves to gain a clearer understanding of the models' performance. Finally, we tested the models by predicting the sentiment of several movie reviews and evaluated the scores for each algorithm.

CHAPTER 2

literature Review

2.1 Sentiment Analysis by IJPREMS, Mumbai

This research paper is focused on analysing Twitter tweets, Twitter is a popular platform to share a real-time information. It highlights how sentiment analysis is crucial for understanding public emotions and reaction on the market trends by analyzing text data. Using machine learning, natural language processing the study shows how we can gain valuable insights from the tweet data and many tweets shared on Twitter. [41]

The main objective of the study is how well sentiment analysis can be done on Twitter data using machine learning and neural network specifically with Long Short-Term Memory (LSTM).

The study involved collecting twitter data, pre-processing or cleaning it to improve the quality of the text, handling missing values and normalizing data feature extraction for training the model. A dataset from kaggle with tweets labeled as positive, negative, and irrelevant. The entire dataset was used to train the model over ten epochs and 32 batch size with validation split of 20%. The LSTM model achieved and accuracy 52.62% on training set and of 55.15% on test set. The paper concludes by noting that these findings can be useful in fields like marketing and public relations. [41]

The strength of the paper include its methods and practical use of neural network. However, the accuracy could be improved, especially since the social media language is complex to clean and various techniques can be used like transformers and artificial neural network

(ANN) for prediction the sentiment. In closing, the study paper effectively illustrates how LSTM models may be used to apply sentiment analysis to Twitter data. It emphasises the value of sentiment analysis in social media interactions and makes the case for more study to increase precision and moral behaviour. Combining sentiment analysis with cutting-edge machine learning methods can help us understand human emotions in online discussions as the field develops. [41]

2.2 Social media sentiment analysis: lexicon versus machine learning

The research paper "Social Media Sentiment Analysis" looks at how well two methods, lexicon-based and machine learning, work for analyzing sentiment in social media. The research paper aims to answer three main research questions stated below:

- Are the existing sentiment analysis techniques suitable for analyzing social media conversation?
- How will the results of lexicon-based and machine learning approaches will differ from each other?
- Will combining both approaches enhance sentiment classification?

The study examines both methods using a large data of consumer content from Facebook brand pages. It measures performance using Precision, Recall, F-score and Accuracy.

Both methods show similar F-scores for classifying positive (0.77 and 0.78) and negative sentiment (0.45 and 0.47). The results challenge past studies that favored machine learning over lexicon based approach. combining both the methods improves the classification of positive sentiment, with an F-score of 0.83. The study concludes that while both methods have their strengths, a combined approach performs better overall, especially for positive sentiment. [8]

2.3 Sentiment Analysis of Social Media Networks Using Machine Learning

The paper "Sentiment Analysis of Social Media Networks Using Machine Learning" studies using different machine learning and deep learning techniques to classify emotions in tweets. The research aims to classify emotions in the large amount of textual data on social media from twitter. This papers aims to find the best techniques from machine learning algorithms and deep learning. The study involves collecting over a million tweets, cleaning the data, and using various classification algorithms like Naive Bayes, decision trees, random forests, neural networks, RNN-LSTM, and CNN, CNN Word2VEC. They also implemented a hybrid model that is a merge of: CNN + CNN (Word2Vec) + RNN (LSTM+Word2Vec). The results show that this hybrid model achieves the highest accuracy of 83.6%, better than traditional methods. [15]

While the paper makes important contributions, there are areas for improvement. The data cleaning steps can be improved like using lemmatization, and checking if there a emojis in the tweets. The hybrid model's structure could be explained better to understanding its working. The study implies that a hybrid approach combining CNN and RNN is very effective for tweets classification. The research finds that hybrid model is much more accurate than traditional methods with highest amongst all. [15]

2.4 Sentiment Analysis for amazon reviews

The authors Eanliang Tan, Xinyu Wang, Xinyu Xu aims to classify positive and negative customer reviews on Amazon products. The dataset has reviews and ratings given my customers on Amazon. Various supervised learning models, traditional algorithms and deep learning methods were created based on this data. The study compares the the accuracy of different models to understand customer emotions about products. Models like Naive Bayes, Linear Support Vector Machines, K-nearest neighbor, Recurrent Neural Networks, and Convolutional Neural Networks were implemented for their performance in sentiment analysis. [14]

The dataset consists of 34,627 Amazon product reviews after cleaning the dataset. The

rating were imbalanced, with class 5 having the most reviews. To overcome this issue the authors over sampled classes 1, 2, and 3. the authors used two types of features : one based on word occurrences and another using 50-dimensional glove dictionary pretrained on Wikipedia. The conventional approach converted reviews into vectors and produced a word dictionary with 4,223 terms. The average vector of individual word vectors was used by the glove dictionary to represent reviews, which reduced the reviews' characteristics and accuracy. [14]

- The dataset was split-ted into training set of 21,000 reviews(60%), a validation set of 6,814 reviews (20%), and a test set of 6,813 reviews (20
- Different models were trained and tested, mostly all of them performed well with traditional input features than Glove input features.
- The Long Short Term Memory (LSTM) model was the most accurate amongst all ranked based on test accuracy.
- The results section highlighted the challenges of data imbalance and over fitting, especially with the resampling techniques used.

All things considered, the study included a thorough examination of feature representation, model performance, and the order of models according to how accurate they were at analysing the sentiment of Amazon reviews. [14]

2.5 Sentiment Analysis of Financial Textual data Using Machine Learning and Deep Learning Models

The research paper involves analyzing and extracting meaningful information like sentiment expressed in a text which can be positive, negative and neutral from financial textual data. The authors of "Sentiment Analysis of Financial Textual Data Using Machine Learning and Deep Learning Models" by Hero O. Ahmad and Shahla U. Umar aims how the application of machine learning (ML) and deep learning (DL) algorithms to perform sentiment analysis on financial texts.[4]

The study uses a mix of ML and DL models to analyze the sentiment in financial texts. The ML model used are Multinomial Naive Bayes (MNB) and Logistic Regression (LR). The

DL models include Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU).[4] The following steps are involved in methodology:

- Data collection: Gathering financial text from various sources
- Preprocessing: Cleaning and preparing the data for analysis
- Model Training: Training the ML and DL models on the prepared data.
- Model Evaluation: Measuring the performance of the models using accuracy metrics

The study finds that both the ML and DL are good at analysing sentiments. The MNB with accuracy score of 74% and LR model with score of 85% have good and very good accuracy. The RNN, LSTM, and GRU models perform even better, with the score of 94%, 96% and 95% showing excellent accuracy. Preprocessing the data improves the models' accuracy.[4]

The paper concludes that the chosen ML and DL models can accurately identify sentiments in financial texts. Preprocessing helps for better performance, and DL models are particularly good at understanding the context and textual data and sequence of the data. the study depicts that ML and DL models both are effective at identifying sentiment classification in financial texts, but the DL are better than ML models. The findings highlight the strong performance of DL models and the importance of data preprocessing.[4]

2.6 Sentiment Analysis Using Product Review Data

The paper "Sentiment Analysis Using Product Review Data" by Xing Fang and Justin Zhan looks at how to understand sentiments in Amazon product reviews as positive, negative, or neutral. The study focuses on understanding and categorizing opinions in text both at the sentence level and the entire review level. [17]

The main question the paper addressing is how effectively the sentiment in amazon product reviews, differs between positive, negative and neutral sentiments. [17]

The authors collected over 5.1 million Amazon product reviews from from four different categories: beauty, books, electronics, and home. Their process involved several steps like, Collecting data and cleaning the reviews to remove irrelevant content. They extracted sentences and performed part-of-speech (POS) tagging to identify words with sentiments, They implemented negation phrase identification algorithm to handle negations, as it is important

for accurate sentiment analysis. They used mathematical models to compute sentiment score for words. For every review, they produced feature vectors based on sentiment scores and additional language characteristics.[17]

The study used three classification techniques: Naive Bayesian, Random Forest, and Support Vector Machine (SVM). The model was evaluated at both the sentence level and the review level. The results showed that:

- The SVM and Naive Bayesian models performed better than the Random Forest model in most cases.
- Both models achieved good results, with F1 scores above 0.8 for sentence-level categorization.
- Review-level categorization also showed strong performance, especially in distinguishing between positive and negative sentiments.

By using a large dataset and robust machine learning models, the study shows effective methods for both sentence-level and review-level sentiment analysis. It suggested that future work could refine these methods and explore additional features to improve accuracy.[17]

2.7 Text Sentiment Analysis Based on Transformer and Augmentation

This paper, "Text Sentiment Analysis Based on Transformer and Augmentation," introduces a new way to analyze sentiments in text by using deep learning transformer models, knowledge distillation, and text augmentation. These methods help tackle challenges like high computational demands and limited labeled data. The main objective of this paper addresses is how to make sentiment analysis models more efficient and accurate while keeping computational costs low and handling situations where there is very little labeled data. [48]

The authors present a model that uses transformers along with two main techniques: knowledge distillation and text augmentation. Knowledge distillation helps reduce the model's size, making it faster and cheaper to run. The transformers used in this study were BERT, ALBERT, and MobileBERT extended version on BERT. [48]

The study was on SST dataset and the AG news corpus. The method proposed in the paper achieved better results with a small number of samples, especially when the number of labeled labels in the AG News dataset was less than 1,000. Notably, when there were only 100 labeled data, the method showed significant improvement compared to other methods . MobileBERT without the mixing layer and data enhancement layer resulted in lower performance compared to other models. The method presented in the paper showed an improvement of about 20% when classifying sentiment analysis and news datasets. The paper highlighted that the proposed method excelled when used with a small amount of labeled data and also performed well with large-scale data. [48]

Tests on two public datasets show that the new model performs as well as other top models. It uses fewer parameters while maintaining high accuracy, making it good for devices with limited computing power and for real-time use. The novel use of both knowledge distillation and text augmentation in transformer models. Reducing computational costs without losing accuracy. Better performance with limited sample sizes. Suitability for real-time use due to faster response times. [48]

This paper significantly advances text sentiment analysis by proposing a model that combines transformer mechanisms with knowledge distillation and text augmentation. These innovations address major challenges related to computational costs and limited labeled data, providing an effective and accurate solution. [48]

2.8 Transformer-based deep learning models for the sentiment analysis of social media data

The paper "Transformer-based deep learning models for the sentiment analysis of social media data" by Sayyida Tabinda Kokab, Sohail Asghar, and Shehneela Naz, explores methods to improve Sentiment Analysis. The authors used new approach that uses Transformer-based models, specifically BERT combined with Convolutional Neural Networks (CNN) and Bi-directional Long Short-Term Memory (Bi-LSTM) networks, to overcome the limitations of traditional models.

The paper looks into how to make models more accurate and efficient when dealing with noisy data context in social media text. the study consists of 4 different dataset: Airline reviews Dataset this dataset consists of 11,517 tweets related to six different United States

(US) airlines, Self-driving car reviews Dataset with 7,156 tweets, US presidential election reviews Dataset containing 10,729 reviews with 21 attributes and IMDB Dataset with over 50,000 movie reviews

The model uses several techniques: Zero-shot classification to label data initially by calculating polarity scores. Pre-trained BERT model to get sentence-level meanings and context. Dilated Convolutional Neural Network (CNN) to capture local and global context in the text. Bi-LSTM to understand long-term dependencies in text sequences.Grid search cross-validation for fine-tuning model parameters. Zero-shot Classification: Using zero-shot classification for initial data labeling is innovative and helps the model handle unlabeled data well.

Integration of BERT: Using BERT to extract semantic and contextual information ensures the model understands the nuanced meaning of words in different contexts.

Dilated Convolution: Applying dilated convolution in the CNN layer allows the model to capture a wide range of contextual information without significantly increasing computational load.

Bi-LSTM for Long-term Dependencies: Including Bi-LSTM ensures the model can understand word sequences in both forward and backward directions, improving its ability to capture long-term dependencies in text.

Comprehensive Evaluation: The model was tested on multiple metrics across different datasets, showing its robustness and versatility.

Findings: The CBRNN model performs better than traditional methods in terms of accuracy, precision, recall, F1-score, and AUC on four different domain text datasets.

The CBRNN model effectively handles noisy data and OOV words while keeping the emotional and contextual information intact, making it ideal for analyzing social media reviews. This paper makes a significant advancement in sentiment analysis by combining Transformer-based models with CNN and Bi-LSTM networks. The CBRNN model effectively addresses the limitations of traditional SA methods, especially in handling noisy data, OOV words, and keeping emotional and contextual nuances. This research provides valuable insights and techniques for improving SA models, highlighting the importance of advanced machine learning approaches in NLP tasks [39]

CHAPTER 3

Methodologies

The section is the key component of this study. Sentiment analysis is one of the major applications of NLP (Natural Language Processing). Here in this section we will discuss detailed methods we will be using in our research.[37] It will help us understand the data Description, pre-processing techniques used for cleaning the data, some visualization for understanding the data and methodologies for predicting sentiments of the movie reviews using Machine learning in detailed manner. The flow chart and the steps taken in this section is shown below in fig 3.1.

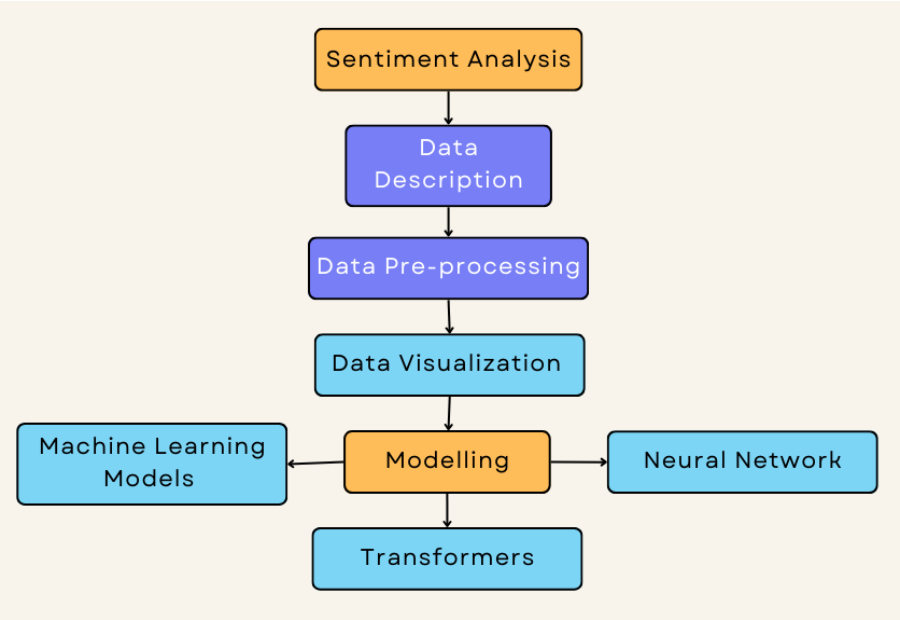


Figure 3.1: Methodology

3.1 Data description

The data **IMDB_datast.csv** -IMDB Movie Reviews Dataset. consists of 'IMDB movie reviews'. IMDB is a site for reviewing movies and is a online movie database. This site consists information about almost all the movies and information about the movies with reviews given by the consumers. The movies are rated from 1-10 according to the popularity. Our data set consists of 50000 movie reviews with their sentiment.

```
shape of data frame: (50000, 2)
```

Figure 3.2: Shape of Dataset

From the above figure 3.2 we can see that our dataset consists of 50,000 rows and 2 columns.

```
Column names:- 'review' and 'sentiment'
Number of reviews:- 50000
Unique sentiment classification:- 'positive' and 'negative'
```

Figure 3.3: Data Description

The dataset **IMDB_datast.csv** consists of two columns naming 'review' and 'sentiment' as shown in figure 3.3, it also consists of 50000 reviews and the sentiment is classified as 'positive' and 'negative'.

In the figure 3.3 below we can see the overview of our dataset **IMDB_datast.csv**

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

Figure 3.4: Data Overview

3.2 Data Pre-processing

Data preprocessing is a crucial step to transform the raw data into a standardized format. This step consists of various steps to clean the data, checking null values, duplicate values and performing necessary transformation on the raw data. For the textual data we will check for null values, duplicate values remove url, convert text to lowercase, remove punctuation, numbers, special characters, removing extra white spaces, stop words, lemmatize the text and checking for emojis in our texts with some visualization to check the outliers. The basic flow of the data pre-processing is shown in the Flow chart below. The main motive of the data pre-processing is to convert the data into desired standard format. Steps involved in text preprocessing is shown in figure 3.4

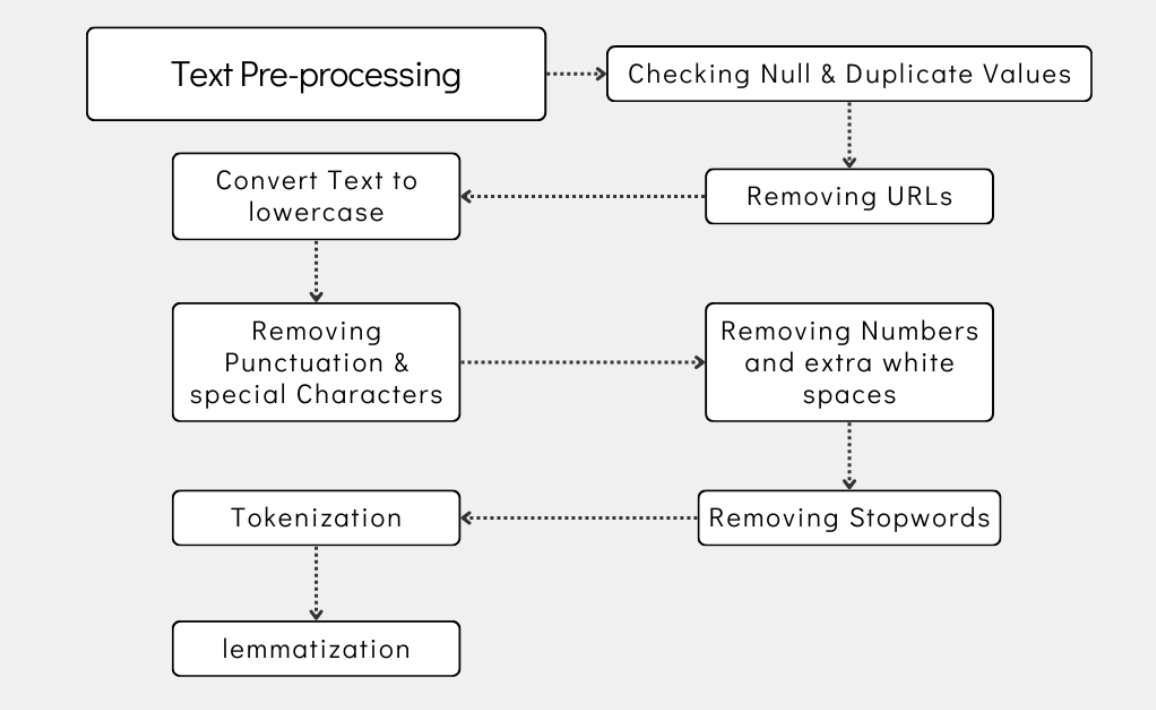


Figure 3.5: Text Pre Processing

3.2.1 Checking null and duplicate values

The first step in the data cleaning is checking for null values, if there are any invalid rows with empty values in our data we will remove it. we will also check for duplicate rows which are repeated more than once, we don't require duplicate rows with same entries repeating more than once this will affect our analysis.

```
Null values in columns:
review       0
sentiment    0
dtype: int64

Duplicate values in the dataset:- 418
```

Figure 3.6: Null and Duplicate Values.

In the figure above 3.5 we can see that our data has no null entries and has 418 duplicate. Therefore, we will remove all the duplicated values from our dataset to improve our performance and accuracy.

3.2.2 Removing URL

When processing the textual data we encounter various text lines which doesn't contributes at all in predicting sentiment, one the major line of text's are URL. "A URL (Uniform Resource Locator) is the address of a unique resource on the internet."[34] URLs won't provide us any insights for our sentiment analysis, we encounter URLs in every form of text line be it a tweet or a post on a social media the URLS are shortened and pasted into the sentences[35]. Therefore, it is better to remove the URLs from the textual data present in our analysis as a first step in our text pre-processing. [35].

3.2.3 Converting Text To Lower Case

Converting the text to lower case is a crucial and important step in text analysis, standardising the text's in a same form helps us for a improved results and accuracy. Lower casing is a important step in 'NLP' tasks for simplicity and helps with consistent output[1]. Some words can change their meaning when changed to lower case[1]. Lower casing also might help in reducing the time taken for prediction, as the data is complex if for example say there are two words 'Happy' and 'happy' the algorithm will consider them as a different entities. But, if we convert the words into lower case the algorithms will consider the word 'happy' after converting it to lower case as a single entity.

3.2.4 Removing Special Characters and Punctuation

During sentiment analysis the textual data contains different types of content in the text like special characters, numbers, extra white spaces and punctuation which are neutral and has no contribution in analysing the sentiment. Removing punctuation and special characters will simplify the text and reduce the noise, punctuation are meaningless and has no sentiment, but it can affect the analysis and probably affect the accuracy[2].

3.2.5 Removing numbers and white spaces

While Analysing the textual data in NLP (Natural Language Processing) one of the most problems are with numbers because numbers do not have specific meaning and can impact our accuracy and results increasing the complexity of our textual data. Textual data contains numbers removing numbers is a crucial steps to reduce the noise of the textual data and decrease the time taken for the algorithms to predict the sentiment of the textual data. Moreover, the extra white spaces in the textual data like for example 'how are you' and 'how are you' carry different meaning during the analysis and is not acceptable during analysis. Therefore, the standard form of the textual data is only with one with space as we all know. So, removing white spaces is and necessary step in our data pre-processing.

3.2.6 Removing Stop words

In NLP (Natural language processing) and text analysis there are words such as "from", "myself", "ours" etc. are considered as filler words that provide minimal informational value[10]. In the context of programming natural language processing (NLP) models and conducting data retrieval, it is necessary to instruct computers to exclude these terms[10]. These non-essential words, which do not contribute substantive meaning, are referred to as stop words[10]. Stop words are prevalent in every human language. Eliminating these words allows us to discard low-level information from our text, thereby emphasizing the significant information [40]. In other words, removing such words does not negatively impact the model's performance for our task[40]. Eliminating stop words significantly reduces the dataset size, which in turn decreases the training time due to the reduced number of tokens involved in the process[40].

We are using 'Wordcloud' library to remove stop words from the textual column in our dataset **'IMDB_dataset.csv'**. There are around 192 stop words we have to remove from the dataset. Some of the stop words are shown below in the figure 3.6.

me, however, http, her's, against, that's, they'll, they'd, such, what, can't, more, them, ill, aren't, doing, but, because, are, hasn't, those, how, let's, your's, myself, we, been, hed, their, im, her, your'e, for, must'nt, i, cannot, they've, have, am, when, i've, com, once, or, has, further, therefore, who's, shall, other, their's, after, themselves, here, also, just, about, where's, dont, why, we've, on, same, she, like, had, you'd, itself, else, out, since, a, ought, ourselves, most, between, id, then, hell, of, does, by, get, when, they're, there, could'nt, some, theres, up, your, that, yourself, is, which, wouldnt, from, until, ever, all, can, otherwise, both, while, the, an, having, you, what's

Figure 3.7: Stopwords

3.2.7 Tokenization

Tokenization in Natural Language Processing (NLP) is the process of breaking down text into smaller pieces called tokens[12]. These tokens can be individual entities or whole words. This process is important because it makes it easier for machines to understand and analyze human language by dividing it into manageable parts [12].

We are using word tokenization in our analysis, Word tokenization breaks text into individual words. It's the most common method and works well for languages like English that have clear word boundaries [12]. for example if we have a text say - "Hello how are you" when you tokenize the sentence each word are identified as a single token like ['Hello', 'how', 'are' ,'you]. We will use NLTK library and Bert tokenizer to tokenize the sentence. NLTK is a well-established Python library in the NLP community that covers various linguistic tasks [12]. It provides both word and sentence tokenization, making it a versatile option for both beginners and experienced users. Building on the BERT pre-trained model, this tokenizer is excellent at context-aware tokenization. It effectively manages the complexities and ambiguities of language, making it ideal for advanced NLP projects[12].

3.2.8 Lemmatization

Lemmatization is a key text pre-processing technique used in natural language processing (NLP) and machine learning, it reduces words to their base forms, called as "lemmas." [20] This technique helps in grouping different forms of a word so they can be treated as a single item and the multiple forms of texts are reduced to less forms[20]. lemmatization considers the context of the words, linking words with similar meanings to a common base form[20]. We are using NLP's 'NLTK' library to process and lemmatize our words. some of the example of lemmatize words are "your's - your", "break, breakes, brokes etc - broke" these are few examples of lemma words[46].

3.3 Data Visualization

Data Visualization is a technique to interpret key insights from the data into visual forms. usually data visualisation is used to find the outliers in the data and for easy understanding of the data and gain meaningful information from the data. there are various types of data visual plots such as bar plot, line plot, density plot, pie chart, histogram, heat-map etc and to interpret these plots we used python libraries named 'matplotlib' and 'seaborn'. This section is really a crucial step also called as Exploratory Data Analysis 'EDA' where we visualize the data and find insights from the dataset given. in this study we are using various techniques to find the outliers and depict necessary information from our **IMDB_datast.csv** .

Distribution of sentiment classification

In the below pie chart 3.8 we can see that in our data with total 50000 reviews the data is split-ed into two categories i.e positive reviews and negative reviews. From 50000 reviews there are 25000 positive reviews and 25000 negative reviews for our analysis. The data is perfectly divided in to half and we can get enough data for training purpose.

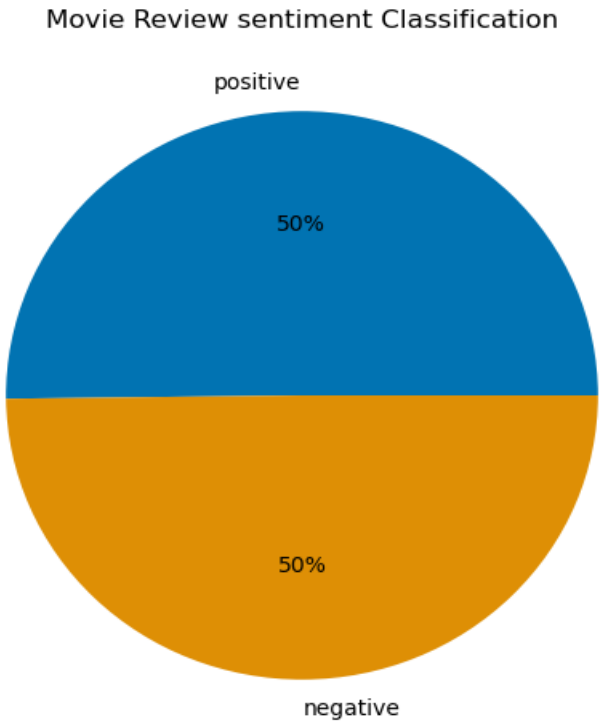


Figure 3.8: Movie Review Classification

Distribution of Text Lengths

In the plot distribution of text lengths the texts are short in the majority of reviews ranging from 0 to 2000 as the text length increases the count of the text decreases. The distribution's form indicates that shorter texts are more prevalent than longer texts.

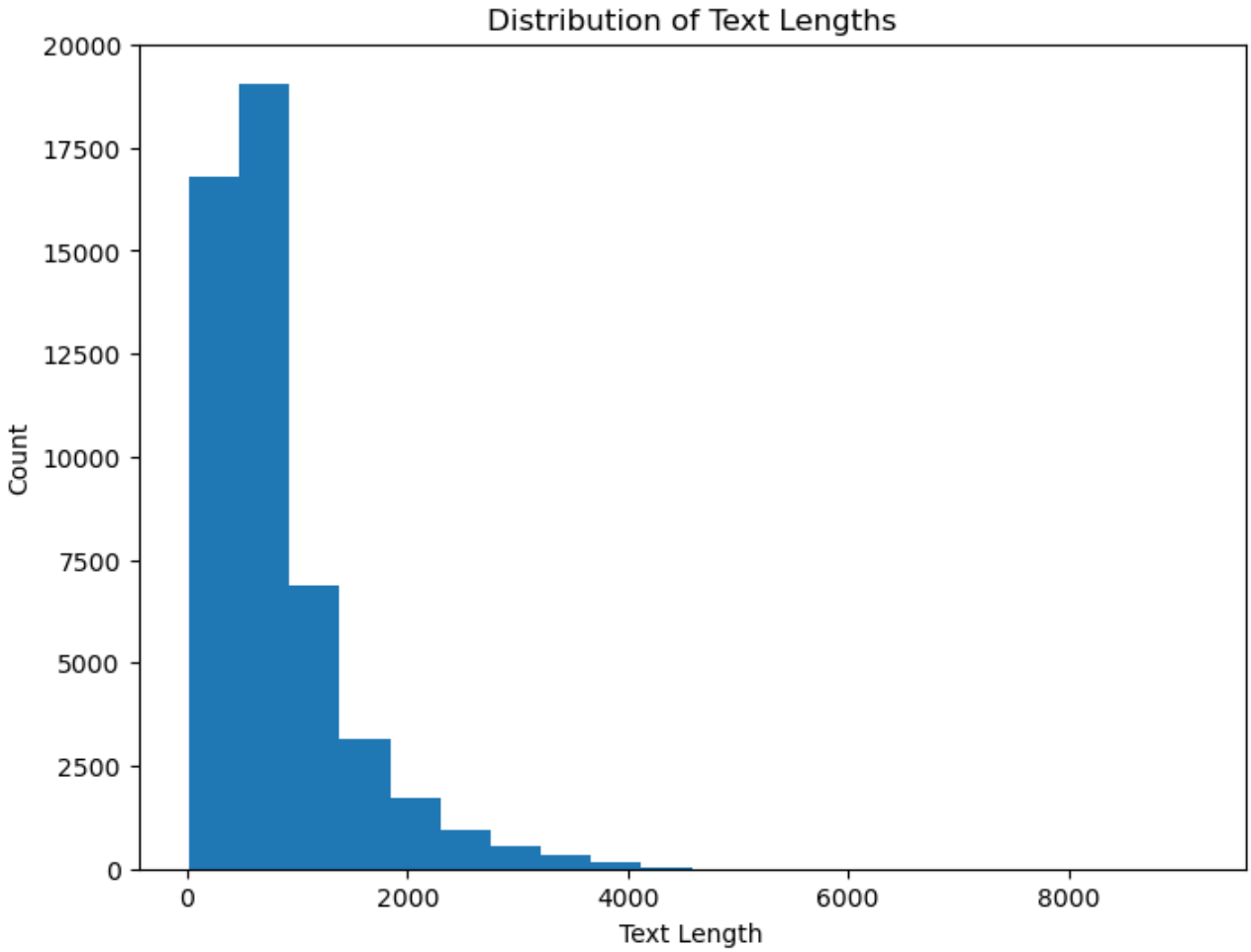


Figure 3.9: Distribution of Text Lengths

Distribution of Review Word Counts

The distribution of review word counts helps us to understand the count of words in the reviews. The words in the reviews are ranging from 0 to 200 words. The reviews with more words are rare meaning most of the review are short. As the length of the review increases the number of words decreases.

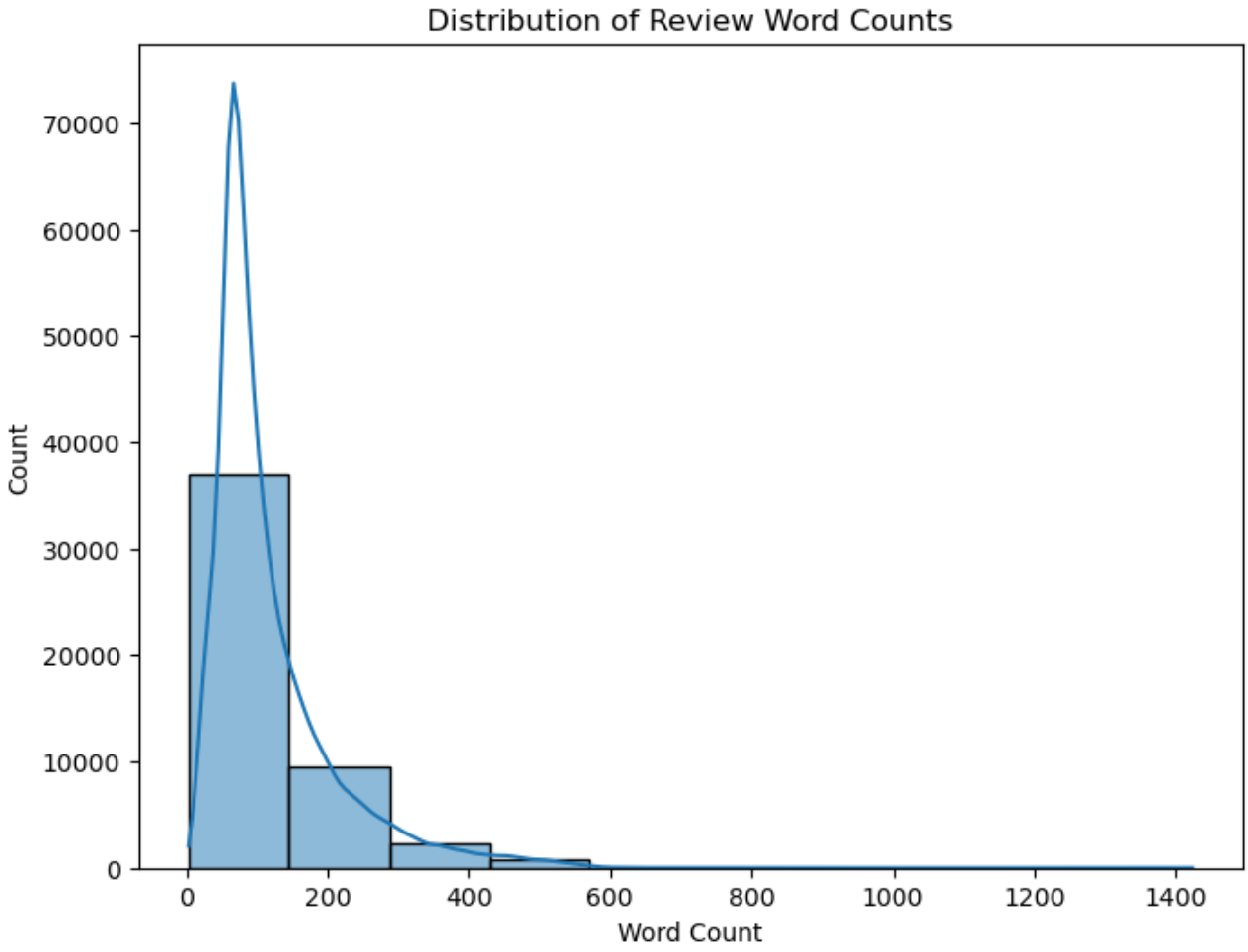


Figure 3.10: Distribution of Review Word Counts

Distribution of Review Lengths

The frequency of short reviews are higher. The reviews in the dataset are short, falling in the range of 0 to 2000 characters. Decreasing Frequency with Length, As the review length increases, the frequency of reviews decreases. The kernel density provides a smooth curve that indicates the density of review lengths across the range. The KDE line peaks sharply at a low character count, reinforcing the histogram's indication that short reviews are the most common. The shape of the distribution suggests that shorter reviews are more common, while longer reviews are less frequent.

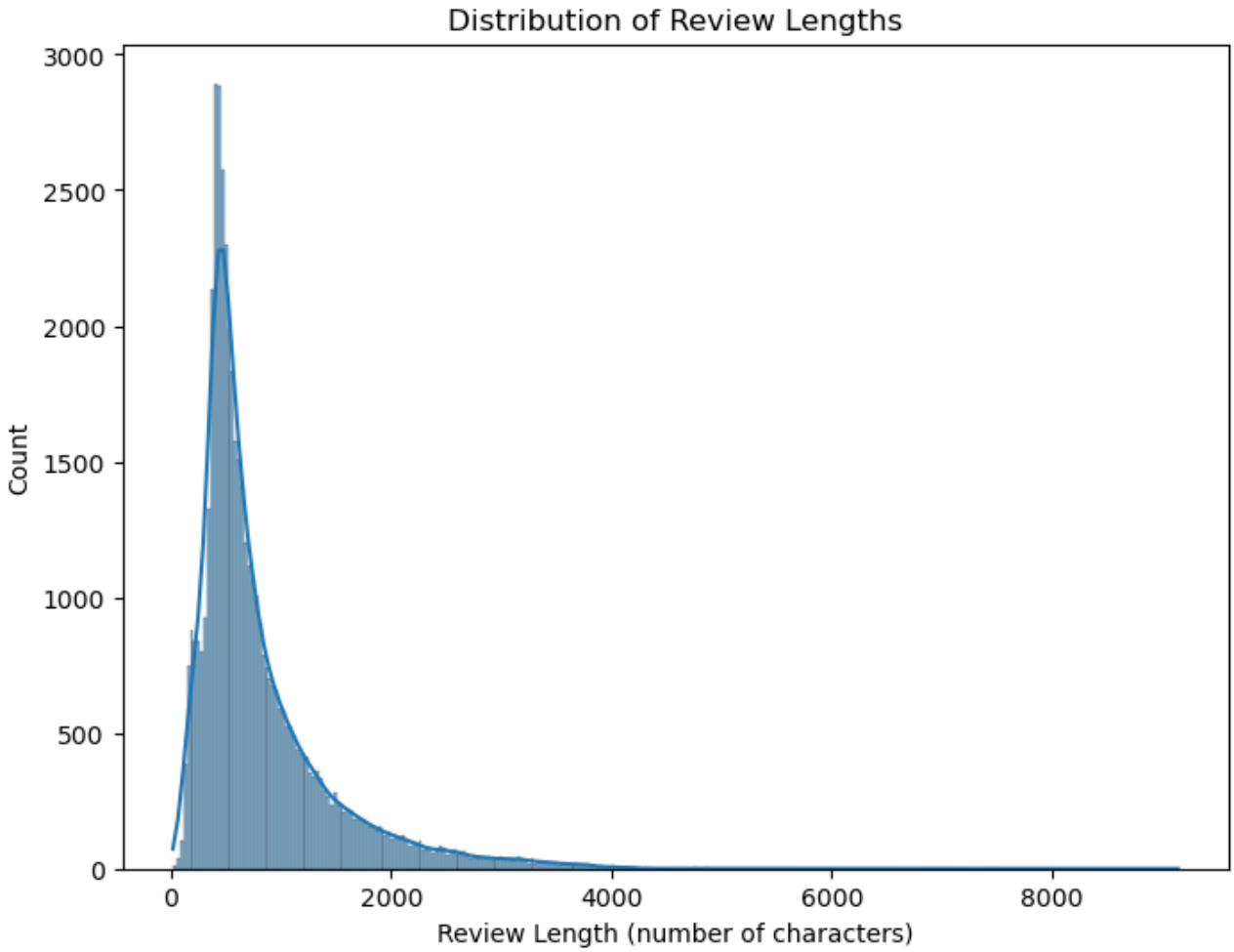


Figure 3.11: Distribution of Review Lengths

Top 20 Words in the data set

In this section we have visualized top words from our movie review. the most top repeated words in the reviews are 'br', 'movie', good', 'time', 'character', 'even', 'story', 'scene', 'well', 'people', 'great', 'bad', 'first' etc. this plot will helped us identify the outliers in the reviews.

Removing outliers

As interpreted in the above bar chart the most repeated word in the movie review is 'br' which is something which is a outlier. 'br' is a break command in HTML which is repeated in the reviews many times. There are such meaningless words in our reviews which we are removing, the words which appeared many times in the dataset and are acting as a outliers in our dataset are ["im","br","go", "got","today","u", "lol","na","amp"]. These words will impact our analysis so removing this word is beneficial for this analysis.

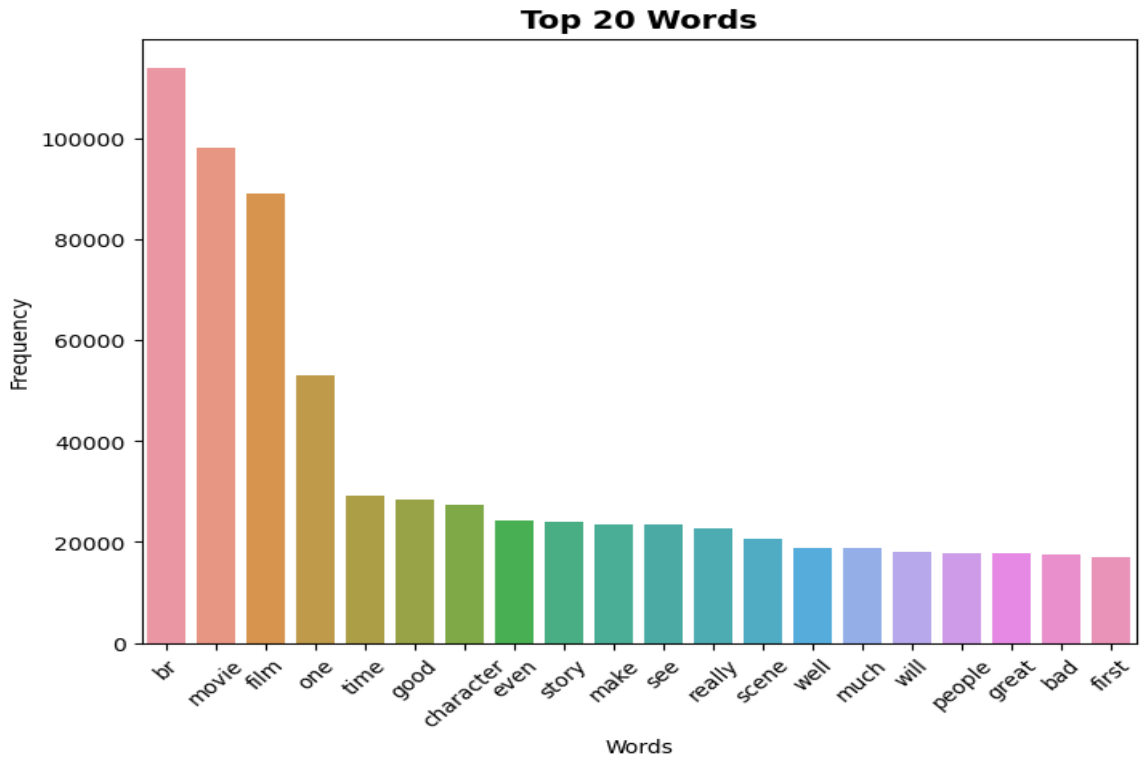


Figure 3.12: Top 20 Words

Top 20 Positive Words in the data set

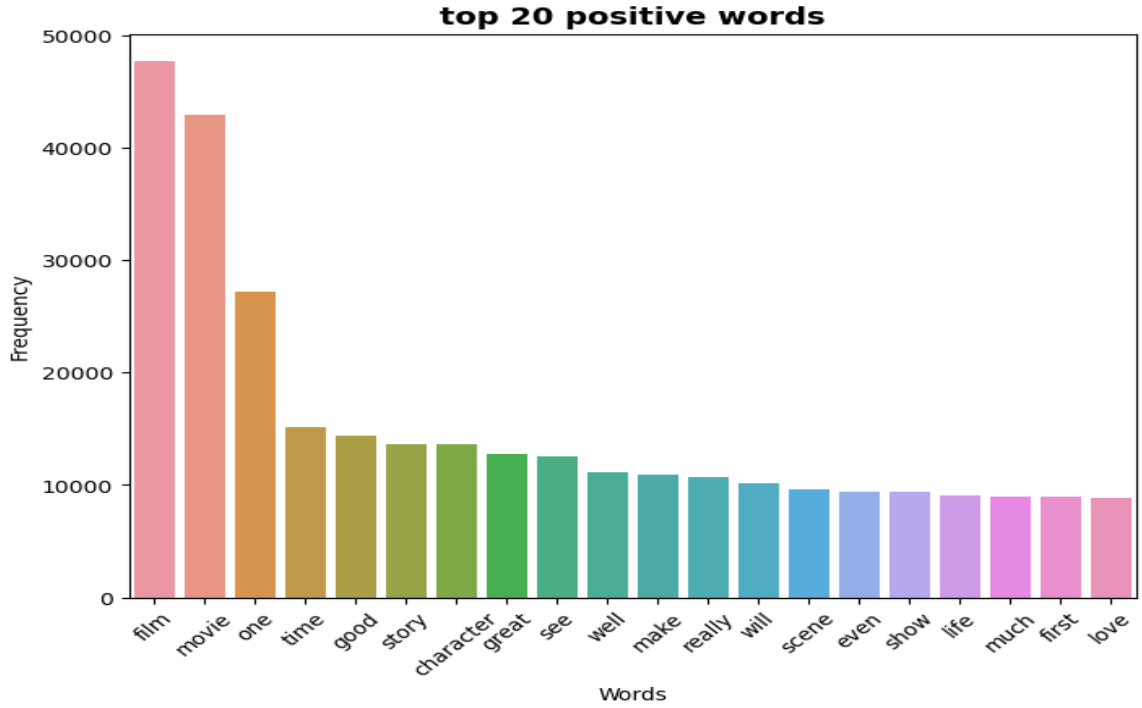


Figure 3.13: Top 20 Positive Words

In the plot we have visualized top words categorized by 'positive' sentiment. We have visualized the words appeared in the positive reviews. The most repeated words in the positive reviews are Film, movie, one, time, good, story,character, great, see, will, scene, even, show, life, much, first, love. The most repeated word in our positive reviews is Film repeated over 48000 times followed by movie which is repeated for about 42000 times.

Top 20 Negative Words in the data set

In the plot we have visualized top words categorized by 'Negative' sentiment. We have visualized the words appeared in the negative reviews. The most repeated words in the negative reviews are Film, movie, one, even, good, time, bad, character, make, really, scene, see, story, don't, much, people, thing, made. The most repeated word in our negative reviews is 'movie' repeated over 55000 times followed by 'film' which is repeated for about 41000 times. the negative words in the dataset are 'bad' ,'dont' etc,

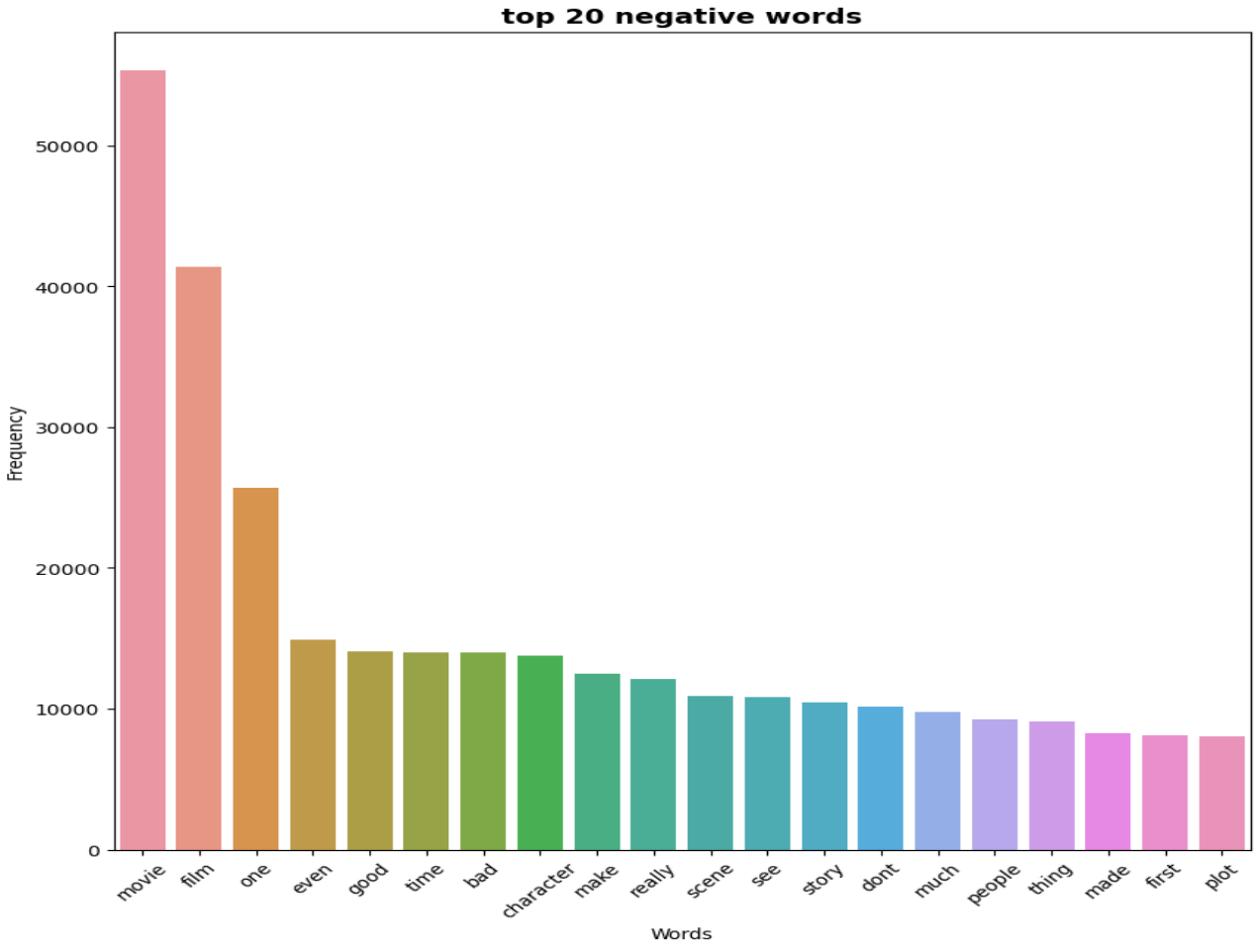


Figure 3.14: Top 20 Negative Words

WordCloud

WordCloud is a visual representation of the words in the textual data. A word cloud is a picture that shows words in different sizes based on how often they appear in a text. Words that are bigger and bolder show up most frequently appeared word and are more important in the text.

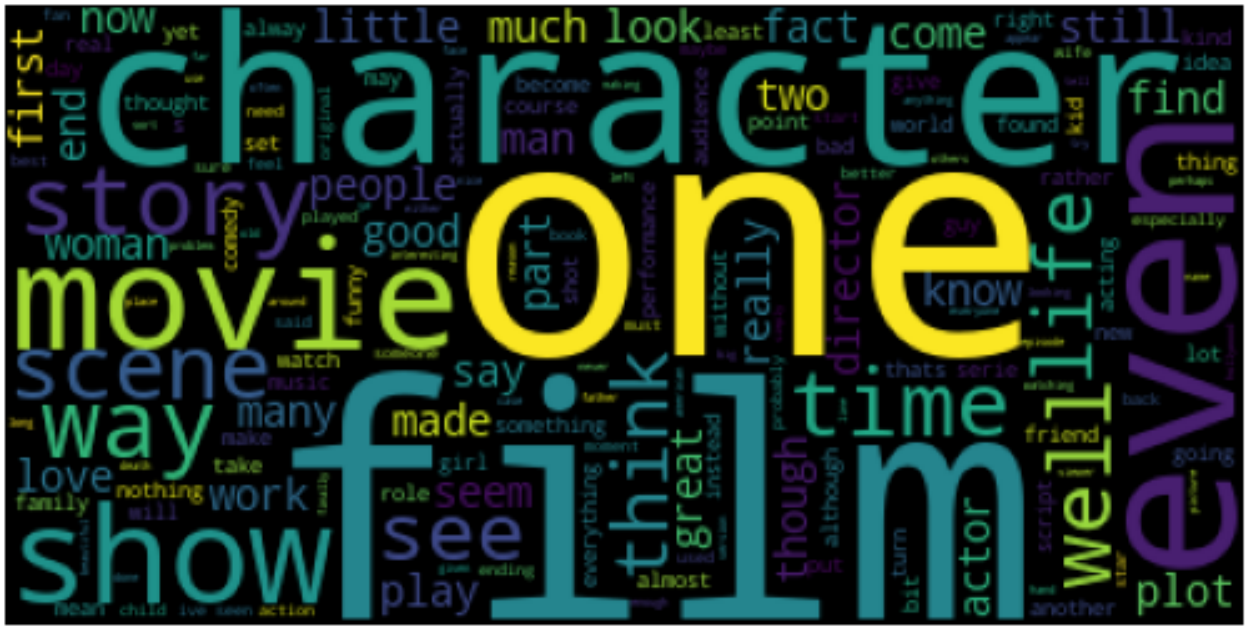


Figure 3.15: Top Words WordCloud

In the above figure we can see depict that the most frequent words in our movie review data are 'Character', 'movie', 'film', 'show', 'life', 'one' etc. using word cloud we can understand the frequency of the words appeared in the movie reviews.

3.4 Modelling

The main goal of this study is to predict the sentiment of the movie reviews whether the review is 'positive' or 'negative'. We have movie reviews as our textual data and we are identifying the sentiment classification of the textual data using Machine learning algorithms for classification. These algorithms can classify if the reviews are positive or negative. We have cleaned the textual data to our desired format removing the unnecessary noise from the data such as special characters, stopwords, and outliers. We, have also used the neural network and transformer models to predict the sentiment we will depict which algorithms and technique is best for predicting the sentiment of the textual data.

3.4.1 Machine Learning Models

In this section we are using the classification algorithms to predict the sentiment of the movie review. The algorithms we used are SVM- support vector machine, Random Forest, Decision tree Classifier, Logistic Regression, Naive Bayes. These are the classification algorithms. A classification algorithm is a supervised learning method that categorizes data into different classes based on training data [16]. This type of predictive modeling is trained with data or observations so that new observations can be sorted into specific classes or groups [16]. The process involves creating a mapping function (f) that links input variables (x) to distinct output variables (y) [16]. like we have to predict the positive and negative sentiment from the movie review [16]. The algorithm produces a probability score for each input, which helps in classifying the data. For instance, email service providers use classification algorithms to assign probability scores to emails, determining whether they belong to the spam category or not [16].

SVM

Support Vector machine is a type of the supervised classification algorithm [9]. SVM classifies the data drawing a line between the classes. Support Vector Machine is a supervised classification algorithm that finds the hyperplane that best separates two classes by maximizing the margin between them as shown in figure[9]. Some of the important terms to be noted in the SVM are:

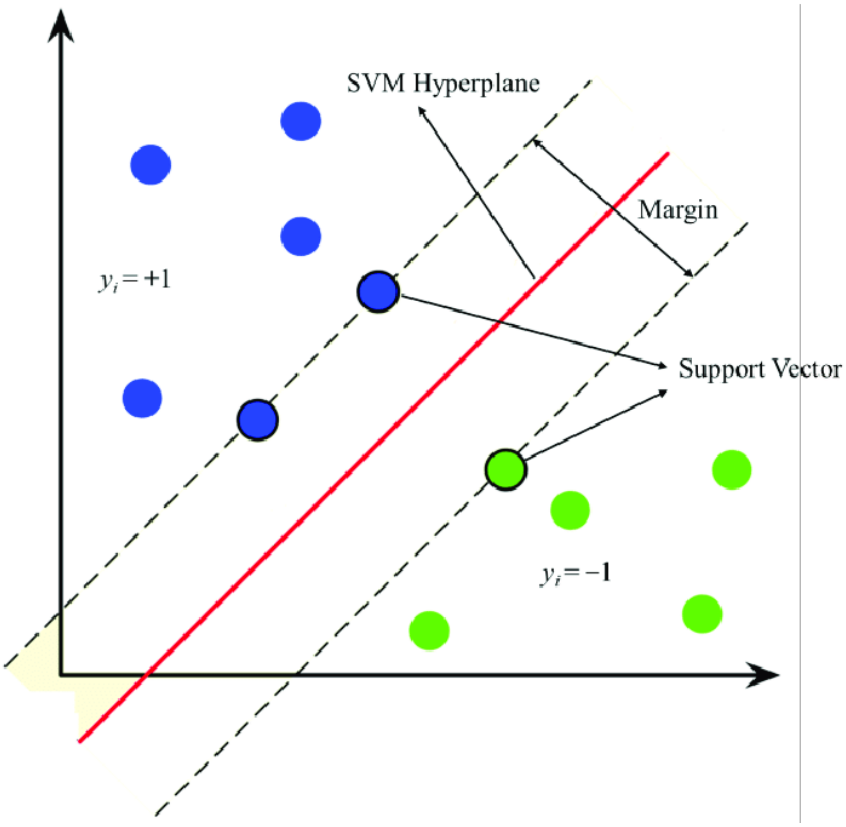


Figure 3.16: SVM (Support Vector Machine [36]

- Hyperplane: In SVM, an hyperplane is used as a decision boundary to separate classes. The classes are separated by a line when the data is two-dimensional, but for a hyper dimension it becomes a hyperplane.
- Support Vectors: Data points that are near to decision boundary
- Margin: Margin is a distance between hyperplane and the data points which are near the hyper plane on any side

Mathematically the SVM [23] is represented as:
The equation of the hyper plane is given by

$$w^T x + b = 0$$

Where 'w' is represents as normal vector to the hyperplane. i.e the direction perpendicular to the hyperplane. 'b' represents the offset or distance of the hyperplane from the origin along the normal vector w.

The distance between data point and decision boundary is denoted as:

$$d_i = \frac{w^T x_i + b}{||w||}$$

Where $\|w\|$ denoted the euclidean norm of w.

Margin is calculated as

$$Margin = \frac{2}{||w||}$$

Where $\|w\|$ magnitude of w.

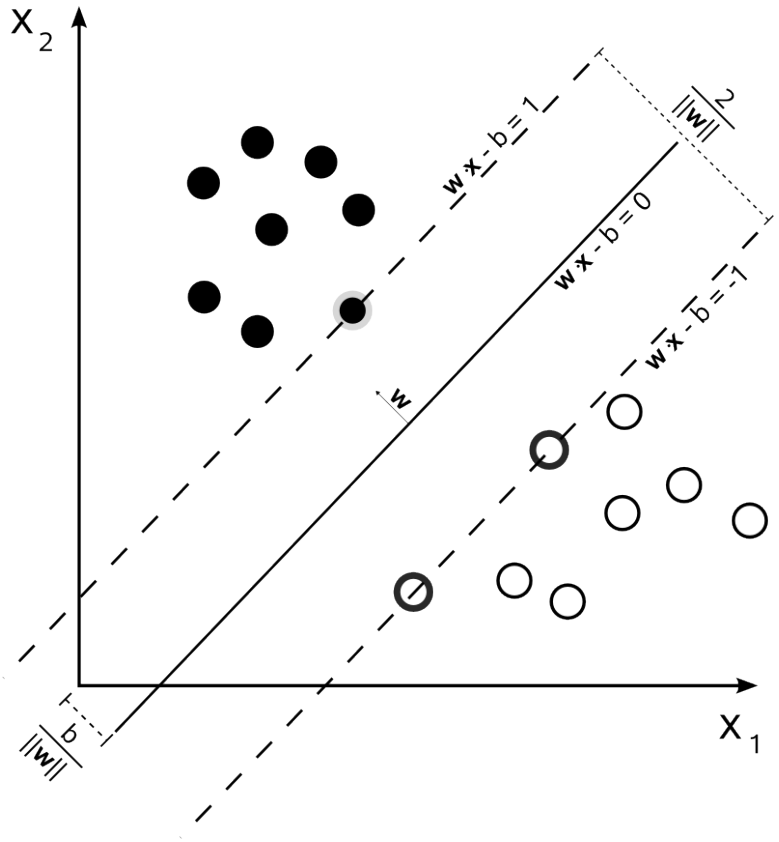


Figure 3.17: SVM Formulation [45]

The SVM helps us to find the sentiment of the movie reviews my classifying it into positive and negative reviews. For training the data the sentiment of the reviews are plotted as the support vectors and for predicting the sentiment the svm find the best hyperplane and predicts the sentiment of the movie review.

Random Forest

Ensemble learning is a technique in machine learning that uses multiple algorithms in machine learning for prediction, it also improves accuracy and performance of the model[3]. The main goal of the ensemble learning model is to boost the strength of the machine learning models, this technique combines multiple results of the machine learning model to reduce errors[3]. types of ensemble learning techniques are - bagging, boosting, random forest etc.

Random Forest algorithms is an ensemble learning method for classification, regression[47]. Random forest built multiple decision trees for training the data and making an decision iterating through multiple decision trees[47]. For classification tasks, like sentiment analysis the output of the random forest is a class selected by the most numbers of decision trees[47].

Random forest has a high predictive accuracy because of multiple decision making, each decision gives a powerful insights predicting a single output, random forest is a team of decision trees. This technique is less prone to over fitting i.e it is resistance to over fitting, suitable for large dataset, and it has a cross validation technique built in it[21].

The random forest algorithm for the sentiment analysis will create a group of decision trees, and the tress are treated and trained on the data. For each movie review the algorithm predicts the sentiment positive or negative. The final outcome is found by measuring the combined amount of the prediction of all the trees.

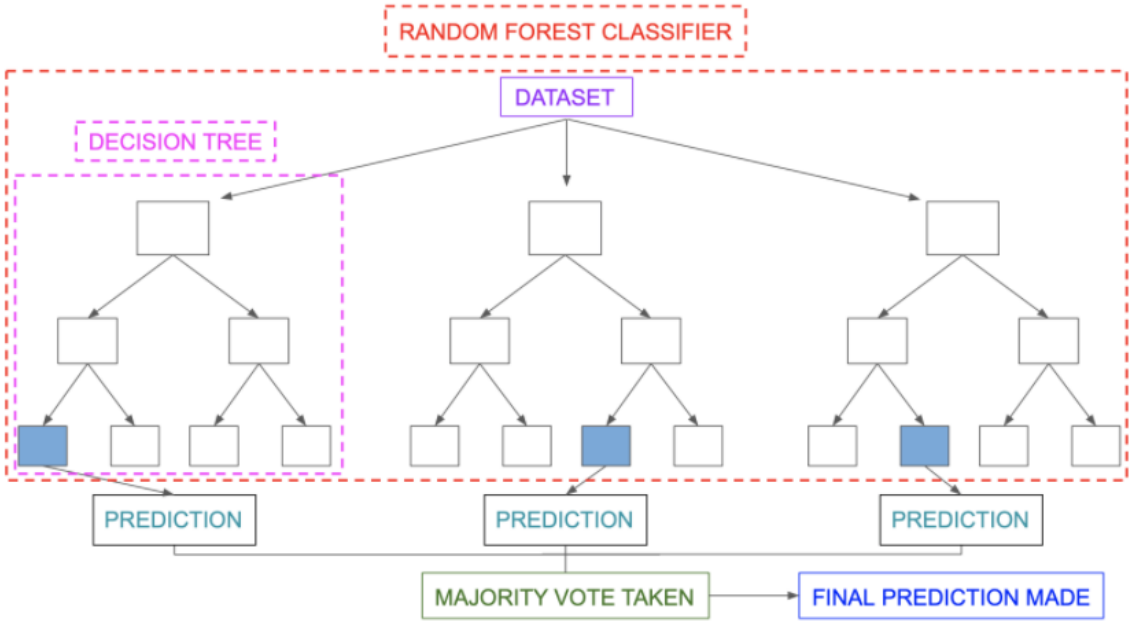


Figure 3.18: Random Forest [32]

Decision tree classifier

Decision tree classifier is a supervised learning algorithm. This algorithm is used for classification purpose. A tree like structure which makes decision based on its potential outcome, a tree and helps with making decisions by showing possible choices and their outcomes[26]. It starts with a root node, then splits into branches that lead to decision nodes and terminal nodes, Where we depict the final results[26].

For sentiment analysis the data is split into different subsets to create a tree like structure[18]. This tree allows to make decisions like predicting the sentiment of the review. each node tell us the sentiment of the review and branches tells us the representation of the decision if the review are positive or negative. Each node tells us the decision point and the branches represent the result of the decision representing if the review is positive or negative. We can evaluate the decision tree model based on various metrics offered my scikit learn library like Accuracy, precision, recall and F1-score[18].

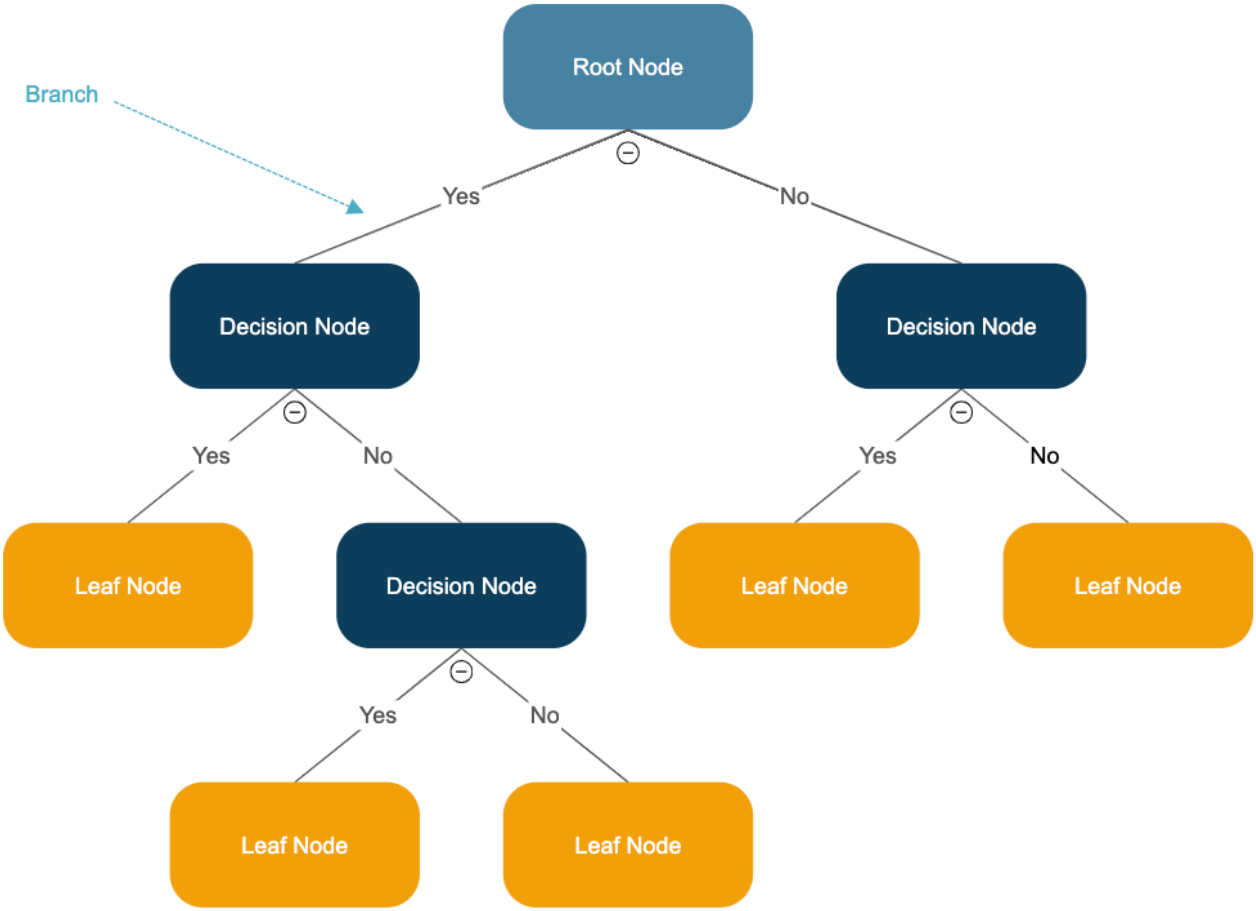


Figure 3.19: Decision Tree [42]

Decision tree algorithms tend to over-fit when they are trained on complex data they also are unstable when minor changes are made in the algorithm the results are affected with large difference[18]. However, Decision tree is easy to understand and flexible to be trained on different types of dataset [18].

Logistic Regression

Logistic regression is used to predict binary results 0 and 1, in our case of sentiment analysis we will predict if the sentiment is positive- 1 and negative- 0. This algorithm has more than one independent variables[5].

Logistic regression mathematical formula is called as Logistic Function.

$$LogisticFunction, P = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}}$$

This function converts various input variables into a probability value within the range of 0 to 1. The objective in logistic regression is to find the optimal coefficients for the input variables that result in the most accurate alignment with the available data[5].

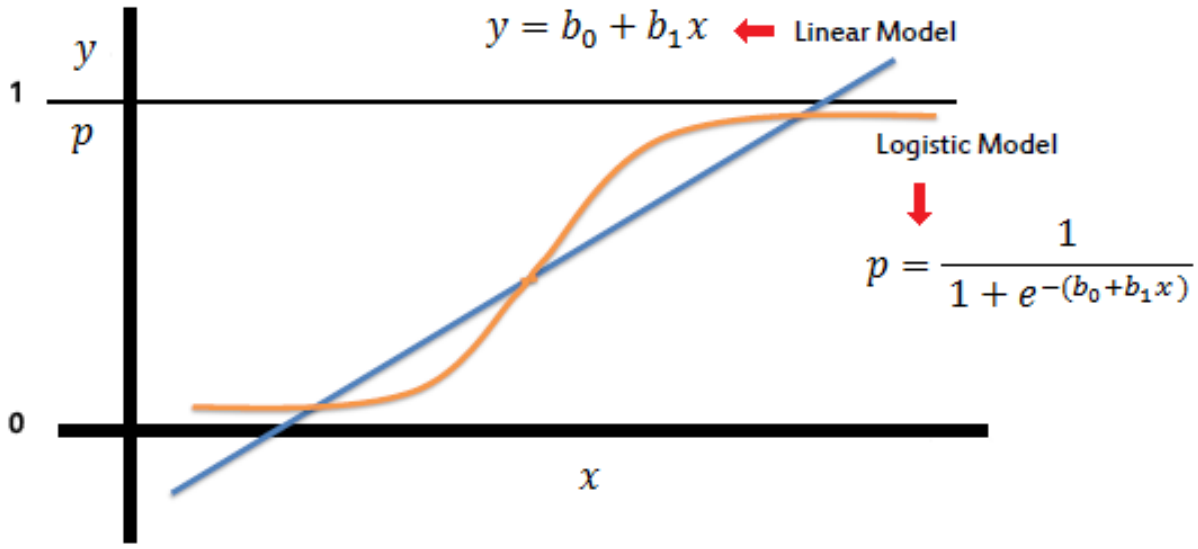


Figure 3.20: Logistic Regression [38]

Logistic regression can also be adapted for more complex situations such as sentiment analysis because sentiment that is textual data has complex structure. For instance, multinomial logistic regression is used here because there are two categories, while ordinal logistic

regression is used for categories with a natural order[5]. Logistic regression are easy to implement and are usually trained fast, less prone to over fitting but can be a issue when can be trained on larger dataset.[5]

Naive Bayes

Naive Bayes is a machine learning algorithm utilized for tasks like text classification, email spam filtering, and sentiment prediction to predict sentiment classes as 0 and 1 [44].

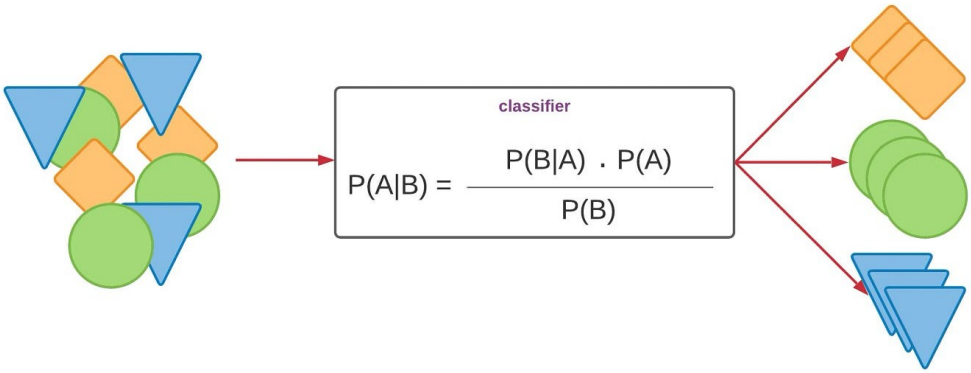


Figure 3.21: Naive Bayes [11]

It is part of a category of algorithms called generative learning algorithms, which aim to represent the distribution of input data for each class or category. Naive Bayes, unlike logistic regression, does not prioritize identifying the most important features for differentiating between classes. Instead, it operates under the assumption that all characteristics are unrelated considering the category, which makes the process uncomplicated, quick, and efficient for forecasting [44].

3.4.2 Neural Network

In simpler terms, a neural network is a type of computer program or method that tries to make decisions in a way that's similar to how the human brain works. It does this by copying how brain cells (neurons) work together to recognize patterns and make sense of things [27]. This refers to a machine learning technique known as deep learning, where layers of interconnected nodes or neurons mimic the structure of the human brain [7]. A Neural network has three layers of connected neurons as followed:

- Input layer: Where all the information to be trained is entered. In the input node the data is processed and is sent to the next layer
- Hidden Layer: The data from the input layer is sent to the hidden layer. Each hidden layer process the data from the previous layer and passes it to next layer. there might be number of hidden layers present in the hidden layer.
- Output Layer: this is a final layer where the output is presented. the output can be in single node or multiple node [7].

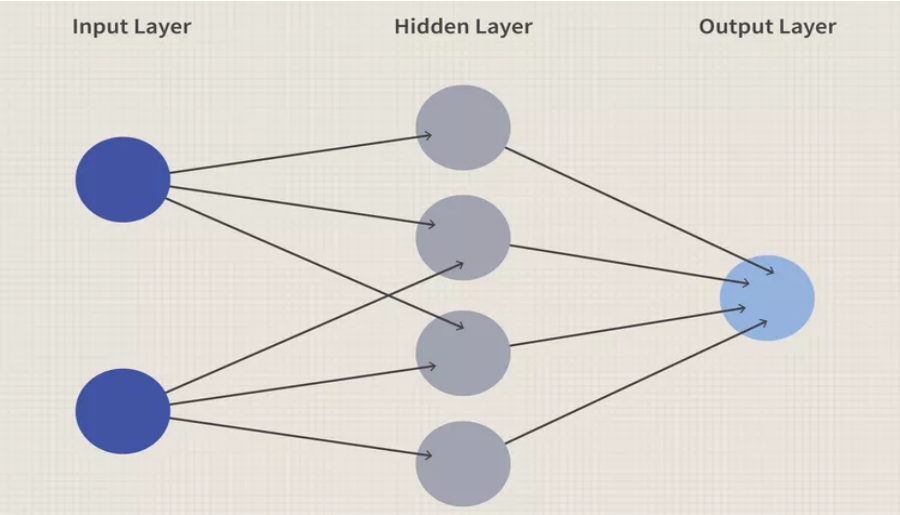


Figure 3.22: Neural Network [29]

In our case the output will be either positive or negative when the textual data will be trained in hidden layer. One of the major application of the Neural Network is Natural Language Processing. The types of neural networks are Convolutional neural networks (CNNs), Recurrent neural networks (RNNs) & Artificial neural network (ANN). WE will be using Artificial neural network for sentiment analysis.

Artificial Neural Network

Ann is a type of neural network and is widely used on textual data and NLP. For the sentiment analysis and its complex textual data ANN is a perfect algorithm to use for predicting the binary classification of the movie review positive and negative. ANN can be use to solve complex structure. To train the model we have used various techniques like tokenization

and padding because neural network requires a fixed layer of input and the length should be same [22].

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 64)	64,000
global_average_pooling1d (GlobalAveragePooling1D)	(None, 64)	0
dense (Dense)	(None, 32)	2,080
dense_1 (Dense)	(None, 16)	528
dense_2 (Dense)	(None, 1)	17

Figure 3.23: ANN Structure

The above figure 3.23 helps us to understand the structure of the ANN, The structure contains 5 layers which includes Embedding, Global Pooling ID and three types of dense layers dense1 , dense2 and output dense layer. The first layer helps us create the vectors of the words for training. This layer plays a vital role in changing the words into the vector forms for continuous representations. The dense layers are used to perform transformations on the input data. Each neuron in a dense layer is linked to every neuron from the layer before it, meaning all neurons are connected to one another. This is why it's called a fully connected layer. [24] The dense layers are the hidden layers in our structure the 1st dense layer takes 64 units from previous layer and the output of that layer is 32 units. the second dense layer decreases from 32 to 16 layers and the last layer is the output layer which has single output for a sentient analysis positive or negative (denoted as 0 or 1).

3.4.3 Transformer

A Transformer is a component of deep learning algorithm and neural network architecture. Transformers convert the input sequences into output sequences. They achieve this by gaining an understanding of the context and monitoring the connections among the elements in the sequence. The main application of transformer is Natural Language Processing. They are used to understanding the human language and predict the outcome or generate new

outcomes. Transformers can be trained on Large scale dataset. One of the best examples of the transformers are CHAT-GPT which uses transformers to predict, answer the questions and other text related tasks. Some of the types of the transformers are BERT- Bidirectional Encoder Representations [6] [28].

BERT model

BERT Transformer - Bidirectional Encoder Representations from transformers, is a type of machine learning model for mainly Natural Language Processing. This transformer was developed by google in 2018 for specifically textual data and to train large textual data, and to improve comprehension of unmarked text for various tasks by training to forecast text both before and after the given text (bi-directional). [25]

BERT transforms words into numerical representations. Machine learning models rely on numerical inputs rather than words, making this process crucial. This enables you to teach machine learning algorithms using your text data. In other words, BERT models are utilized to convert your text data for utilization in a ML model to make predictions alongside other data types [25].

BERT model Architecture

The Figure below 3.24 represents the structure of the Structure of BERT transformer. The BERT models has layers which include embeddings, encoder, bertpooler, dropout and classification layer. The first embedding layer converts words into tokens same as neural network because the data cannot be trained on textual data in our case the dimension of the word embedding is 768. The main part of the structure are encoders these are the stack of 12 layers of transformers. [13] The initial token ([CLS] token) is extracted after the encoder layer. For classification tasks, this unique token is intended to stand in for the complete sequence. A Tanh activation function and a dense layer are applied to the embedding of the [CLS] token. The representation of a sentence is this pooled output. [13] the next part is the dropout layer to avoid overfitting during training, a dropout layer with a dropout rate of 0.1 is added to the pooled output. The final layer is a classification layer to predict the outcome of the movie review. [13]

```
BertForSequenceClassification(
  (bert): BertModel(
    (embeddings): BertEmbeddings(
      (word_embeddings): Embedding(30522, 768, padding_idx=0)
      (position_embeddings): Embedding(512, 768)
      (token_type_embeddings): Embedding(2, 768)
      (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
      (dropout): Dropout(p=0.1, inplace=False)
    )
    (encoder): BertEncoder(
      (layer): ModuleList(
        (0-11): 12 x BertLayer(
          (attention): BertAttention(
            (self): BertSdpaSelfAttention(
              (query): Linear(in_features=768, out_features=768, bias=True)
              (key): Linear(in_features=768, out_features=768, bias=True)
              (value): Linear(in_features=768, out_features=768, bias=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
            (output): BertSelfOutput(
              (dense): Linear(in_features=768, out_features=768, bias=True)
              (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
              (dropout): Dropout(p=0.1, inplace=False)
            )
          )
          (intermediate): BertIntermediate(
            (dense): Linear(in_features=768, out_features=3072, bias=True)
            (intermediate_act_fn): GELUActivation()
          )
          (output): BertOutput(
            (dense): Linear(in_features=3072, out_features=768, bias=True)
            (LayerNorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
            (dropout): Dropout(p=0.1, inplace=False)
          )
        )
      )
    )
    (pooler): BertPooler(
      (dense): Linear(in_features=768, out_features=768, bias=True)
      (activation): Tanh()
    )
  )
  (dropout): Dropout(p=0.1, inplace=False)
  (classifier): Linear(in_features=768, out_features=2, bias=True)
)
```

Figure 3.24: Transformer Structure

3.4.4 Evaluation Methods

To evaluate the model and find the precise and accurate model we have used some evaluation metrics from 'sklearn' library in python we have evaluated accuracy,precision, recall and f1-score of the models to find the best performing model. In some cases we have also used training and testing acuracy to evaluate the model. To interpret the best model from all the models we have plotted confusion matrix of the model and ROC-curve to interpret the findings from the performed models.

Confusion Matrix

A confusion matrix is a tool used to evaluate how well a machine learning model performs on test data. It helps us see how many correct and incorrect predictions the model made. This matrix is especially useful for models that classify data into different categories. [19][43] The confusion matrix shows the following:

- **True Positive (TP):** The model correctly predicted a positive outcome.
- **True Negative (TN):** The model correctly predicted a negative outcome.
- **False Positive (FP):** The model incorrectly predicted a positive outcome This is also called a Type I error.
- **False Negative (FN):** The model incorrectly predicted a negative outcome. This is also called a Type II error.

A confusion matrix is important when evaluating how well a classification model works. It provides a detailed breakdown of true positives, true negatives, false positives, and false negatives, helping us understand the model's recall, accuracy, precision, and overall ability to distinguish between different classes. This matrix is particularly useful when the data has an uneven distribution of classes, allowing for a more in-depth evaluation than just looking at basic accuracy. [19][43]

Accuracy

Accuracy is a way to see how good a machine learning model is at making correct predictions. It tells us what percentage of the total predictions the model got right from the training dataset or test dataset. [19][43] Accuracy is mathematically represented as follows:

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

Precision

Precision depict how many of the predictions the model made for the positive class were actually correct i.e positive values. [19][43] Precision is mathematically represented as follows:

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

Recall

Recall tells us how good the model is at finding all the positive cases. High recall of the model means the model is good at predicting all the positive cases. [19][43] Recall is mathematically represented as follows:

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

F1 score

F1 Score is a single number that combines precision and recall, this combine metrics predicts the model performance. making it easier to see how well the model balances these two. It's especially useful when both precision and recall are important. F1 Score is mathematically represented as follows: [19][43]

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

ROC curve

The ROC curve is a graph that shows how well a model can separate positive cases from negative ones as the threshold for making predictions changes. It looks at two things: the true

positive rate (which is the same as recall) and the false positive rate (how often the model wrongly predicts a negative case as positive). [19][43]

AUC score

AUC (Area Under the Curve) is a number that summarizes the ROC curve. It tells us how well the model can tell the difference between positive and negative cases overall. [19][43]

CHAPTER 4

Results

In this study we have successfully implemented various techniques to predict the movie reviews. We have used Machine learning algorithms, neural network and transformer to predict if the review is positive and negative. To evaluate the model we have used metrics from scikit learn library. We will evaluate the model using various metrics like Accuracy, Precision, Recall, F1-score, Training accuracy, Testing Accuracy and few plots for easier understanding. We have analysed the data and predicted the outcome of the movie reviews.

4.1 Machine Learning Algorithm Result

In this section we have evaluated various machine learning algorithms used in this study in the table below 4.1. We have used Support Vector Machine, Logistic Regression, Decision Tree, Random Forest and Naive Bayes.

Machine Learning Algorithms	Accuracy	Precision	Recall	f1-score
Support Vector Machine	85.38	84.01	87.54	85.74
Logistic Regression	88.70	87.48	90.43	88.93
Decision Tree	71.36	71.67	71.01	71.34
Random Forest	84.37	84.12	84.87	84.50
Naive Bayes	86.34	86.53	86.21	86.37

Table 4.1: Result of Machine learning algorithms

All of these algorithms are classification algorithms which we used to predict the sentiment of the movie review. In this all the metrics tells us how we predicted the sentiment of the movie review which is our primary goal of the project.

As shown in the table Logistic regression performs best amongst all the other machine learning algorithms. Considering Accuracy 88.70% , 87.48%, recall 90.43% and F1-score 88.93% suggest that the logistic regression algorithm have successfully identified the positive and negative sentiments balancing precision and recall. The second best is naive bayes with accuracy 86.34% , 86.53%, recall 86.21% and F1-score 86.37%. Random forest and support vector machine performed decently but couldn't outperform logistic regression. The worst performed algorithm in decision tree amongst all the machine learning algorithms.

The logistic regression model performs best in machine learning for sentiment prediction. its recall is 90.43% which is the best amongst all predicting 90% of actual positive sentiments.

4.2 Artificial Neural Network Result

The Neural Network Model Artificial Neural Network is evaluated below in the table 4.2

Table 4.2: Metrics of Artificial Neural Network algorithms

Model	Training Accuracy	Testing Accuracy	Accuracy	Precision	Recall	f1-score
ANN	86.67	84.69	85.86	83.86	86.31	85.07

The artificial neural network give us accuracy of 86.67%, 84.69% accuracy on training data and test data, and the accuracy of 85.86% these metrics tells us how well the model performed on training and test dataset. Other metrics such as Precision 83.86% Recall 86.31% and f1-score 85.07% gives us an idea of how well the model predicted actual true positive classification, ratio of actual positive reviews which were identified correctly. To improve the accuracy of the neural network we have performed some basic hyper parameter turning and the results are presented in the table below 4.3

Table 4.3: Metrics of Tuned Artificial Neural Network algorithms

Model	Training Accuracy	Testing Accuracy	Accuracy	Precision	Recall	f1-score
ANN	90.27	84.75	84.77	82.05	88.61	85.21

The neural network model ANN performance isn't good as machine learning model even after performing hyper-parameter tuning. To fine tune the model we increased the vocabulary size from 1000 to 2000 if we increase it more the model will over fit, moreover we also increase complexity of our model changed the activation function from 'tanh' to 'relu', added dropout layer to provide , adjusted the learning rate as the default rate might not be perfect for training. Although, we have tuned the model we don't see a major change in results. There is a slight increase of accuracy of 1% which is minimal. The model performance isn't good as tradition models even after tuning it. The reasons for the less accuracy of neural network over traditional model might be that the data which we are using might not be complex enough because neural network is suitable for complex datasets for its high performance. If the dataset is small for the neural network model they tend too over fit, we can depict this by checking the difference between the training and testing accuracy of the tuned model there is a difference of 4% and some over fitting is noted.

4.3 BERT Transformer Result

IN the table below 4.4 we have depicted the metrics for sentiment analysis of BERT transformer. we are predicting if the movie reviews are positive or negative based on the textual data.

Table 4.4: Result of BERT Transformers

Model	Accuracy	Precision	Recall	F1-score
BERT Transformer	89.86	88.58	91.10	89.82

The accuracy of BERT transformer is 89.86 which is higher than all the algorithms we used in our study. this accuracy denotes that the model identifies the positive and negative sentiments effectively. The precision of 88.58 tell us how effectively our model predicted true values i.e positive sentiments and as we can see our result has lesser false positive in transformer model than other models. The recall and the f1-score of the model is 91.10, 89.82 percent which is really good out of all the models and tells us how effectively the model predicted actual positive reviews and is great at predicting positive reviews, and the f1-score helps us to understand the overall performance considering precision and recall which is 89.82% which suggest best overall performance with high precision and recall.

4.4 Best Model Selection

We have successfully implemented the techniques to classify the movie review sentiment and predicting if the reviews are positive or negative. In this section we will compare all the models and techniques performed in our study.

4.4.1 Comparison of models

We have interpreted Machine learning, neural network and transformer model in our study in the table below .

Table 4.5: Metrics of all the Models

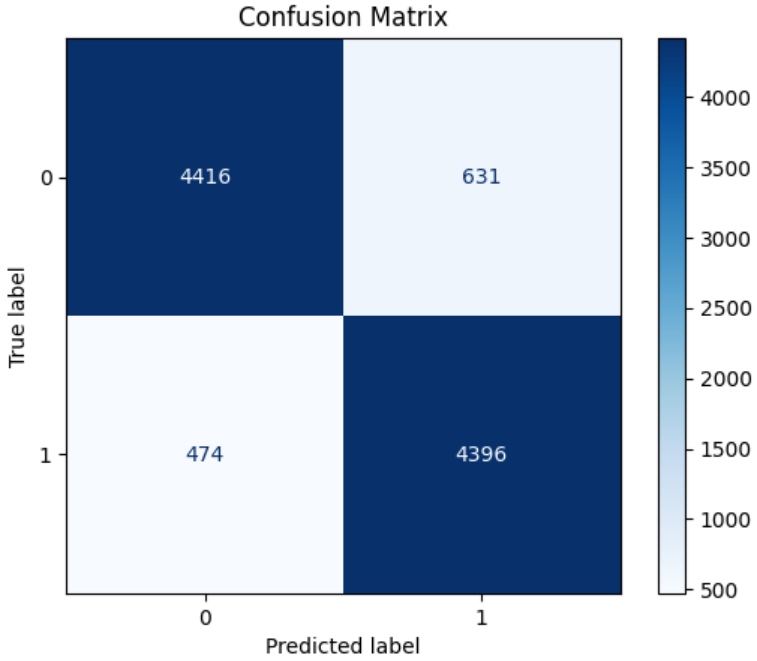
Models	Accuracy	Precision	Recall	F1-score
Support Vector Machine	85.38	84.01	87.54	85.74
Logistic Regression	88.70	87.48	90.43	88.93
Decision Tree	71.36	71.67	71.01	71.34
Random Forest	84.37	84.12	84.87	84.
Naive Bayes	86.34	86.53	86.21	86.37
ANN	85.86	83.86	86.31	85.07
Tuned ANN	84.77	82.77	88.61	85.21
BERT Transformer	89.86	88.58	91.10	89.82

AS we can see in the above table we have interpreted all the accuracy, precision, recall and f1-score of all the models used in this study. The data is trained on 80% of the data and tested on 20% of the data. From the table we can analyse that the machine learning algorithms performs really well and more than expected.

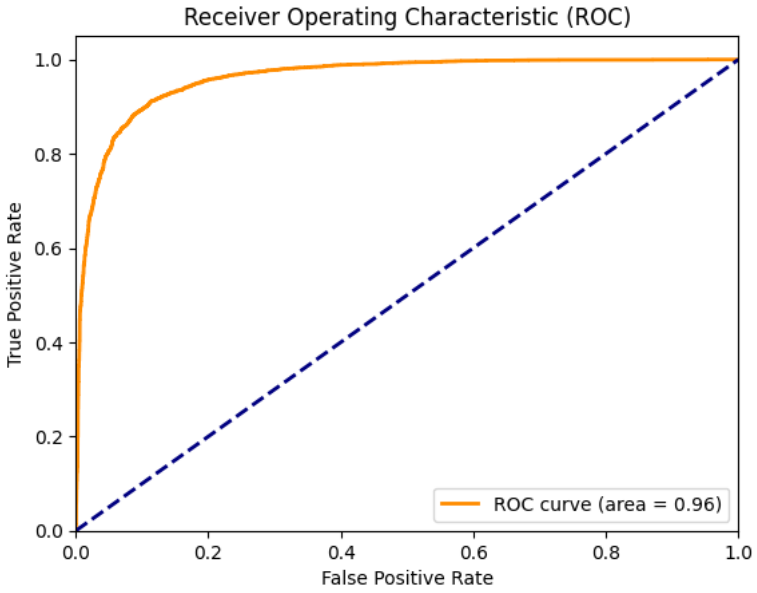
The machine learning model which performed good and were good at predicting the sentiments are Support vector machine and logistic regression with almost 88% accuracy. However, the neural network model under performed and most of the ML models outperformed neural network model. The best performed model with higher accuracy, precision, recall and f1-score is transformer amongst all the models in this study. Transformers has Better accuracy than best model of ML in predicting sentiments. The overall performance of the transformer is the best if we consider all the metrics.

4.4.2 Plotting Confusion matrix and ROC curve of Best Model

In the above section we have selected the best model which is transformer model and based on transformer model, the confusion matrix and ROC curve is plotted in the figure 4.1 shown below.



(a) Transformer Confusion Matrix



(b) ROC Curve

Figure 4.1: Transformer Confusion Matrix and ROC curve

Shown from the confusion matrix 4.1 above, it is understood that the model has classified 4475 true negative and 4437 as true positive, i.e the model has predicted 4475 negative sentiments correctly and 4437 as a positive reviews accurately. The transformer gives us a ROC score 0f 0.96 i.e the model is classifying the positive and negative movie reviews using 96% of the instances, showing the best amongst all the models. The Transformer model is the best model overall considering the metrics and the performance of the model. Moreover, transformer has its own advantages over other model because the transformers are particularly made for Natural Language Processing i.e for textual data and can handle complex data files and long sentences so considering all the pros and cons and evaluation the transformer is the best model.

CHAPTER 5

Conclusions

In the new advanced technological era and vast data we come across textual data in our daily life, be it social media, work, e commerce, health care,etc with the advancement in the field of technology Natural Language Processing has played a crucial role in this industrial advancement. Working on textual data is really vast and complex because of unstructured data and many outliers present in each text including emojis, special characters etc which carries no meaning in some cases. Thanks for NLP a branch of Deep learning where we can perform various analysis on textual data and find insights on data.

In our study we have successfully performed various techniques and algorithms to predict a sentiment of a movie review classified as positive and negative sentiments. We have successfully use some machine learning algorithms, neural network and transformer based model to predict the sentiment and reached to a conclusion that the transformer is a best model to predict a sentiment and can be trusted to predict sentiments of a textual data in any domain and more classification. This model will help us in many way to understand the textual data in every possible way for instance we can predict type of cancer using the symptoms, we can tell the stock trend using the news and recent company financial etc. This research has successfully predicted the sentiment and compare the different technology including traditional and latest technology. In conclusion the best model is the transformer model with 90% of accuracy for predicting sentiments and is best that traditional models.

Bibliography

[1] Text preprocessing for nlp and machine learning tasks. https://resources.experfy.com/ai-ml/text-preprocessing-for-nlp-and-machine-learning-tasks/#:~:text=Text%20often%20has%20a%20variety,with%20consistency%20of%20the%20output., 2023. Accessed: 2024-07-26.

[2] Text preprocessing for nlp and machine learning tasks. https://resources.experfy.com/ai-ml/text-preprocessing-for-nlp-and-machine-learning-tasks/#:~:text=Text%20often%20has%20a%20variety,with%20consistency%20of%20the%20output., 2023. Accessed: 2024-07-26.

[3] ActiveLoop. Ensemble learning, 2024. Accessed: 2024-08-03.

[4] Hero O. Ahmad and Shahla U. Umar. Sentiment analysis of financial textual data using machine learning and deep learning models. *ResearchGate*, 2021.

[5] Algogene. Community post: Decision trees, 2024. Accessed: 2024-08-03.

[6] Amazon Web Services. What are transformers in artificial intelligence? Amazon Web Services website, 2024. Accessed: 2024-08-10.

[7] Amazon Web Services. What is a neural network? Amazon Web Services website, 2024. Accessed: 2024-08-10.

[8] Cynthia M. Webster Chedia Dhaoui and Lay Peng Tan. Social media sentiment analysis: Lexicon versus machine learning. *Journal of Consumer Marketing*, 34(3):218–229, 2017. Accessed: 2024-07-08.

[9] Hands-On Cloud. Svm python tutorial, 2024. Accessed: 2024-08-03.

[10] Coursera. What are stop words?, 2022. Accessed: 2024-07-26.

[11] Dancerworld60. Demystifying naÃ¯ve bayes: Simple yet powerful for text classification. *Medium*, 2024. Accessed: 2024-08-05.

[12] DataCamp. What is tokenization?, 2024. Accessed: 2024-07-26.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[14] Xinyu Xu Eanliang Tan, Xinyu Wang. Sentiment analysis for amazon reviews, 2018. Accessed: 2024-07-08.

[15] Mohammed H. Abd El-Jawad, Rania Hodhod, and Yasser M. K. Omar. Sentiment analysis of social media networks using machine learning. In *2018 14th International Conference on Innovations in Information Technology (IIT)*, pages 85–90. IEEE, 2018. Accessed: 2024-07-08.

[16] Emeritus. Artificial intelligence and machine learning: Classification in machine learning, 2024. Accessed: 2024-08-03.

[17] Xing Fang and Justin Zhan. Sentiment analysis using product review data. *Springer Link*, 2015.

[18] FasterCapital. Decision trees: Decoding decision trees for accurate predictive modeling, 2024. Accessed: 2024-08-03.

[19] GeeksforGeeks. Confusion matrix in machine learning. https://www.geeksforgeeks.org/confusion-matrix-machine-learning/, 2024. Accessed: 2024-09-01.

[20] GeeksforGeeks. Python lemmatization with nltk, 2024. Accessed: 2024-08-03.

[21] GeeksforGeeks. Random forest algorithm in machine learning, 2024. Accessed: 2024-08-03.

[22] GeeksforGeeks. Rnn for text classifications in nlp. GeeksforGeeks website, 2024. Accessed: 2024-08-10.

[23] GeeksforGeeks. Support vector machine algorithm, 2024. Accessed: 2024-08-05.

[24] GeeksforGeeks. Difference between an embedding layer and a dense layer. https://www.geeksforgeeks.org/difference-between-an-embedding-layer-and-a-dense-layer/, n.d. Accessed: 2024-09-15.

[25] H2O.ai. Bert. H2O.ai Wiki, 2024. Accessed: 2024-08-10.

[26] Heavy.AI. Decision tree analysis, 2024. Accessed: 2024-08-03.

[27] IBM. Neural networks. IBM website, 2024. Accessed: 2024-08-10.

[28] IBM. Transformer model. IBM website, 2024. Accessed: 2024-08-10.

[29] Investopedia. Neural network. Investopedia website, 2024. Accessed: 2024-08-10.

[30] Bing Liu. *Sentiment Analysis and Opinion Mining*, volume 5 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers, 2012.

[31] Andrew L. Maas, Ray E. Daly, P. T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150. Stanford AI Lab, 2011.

[32] Medium. Image of a decision tree diagram, 2024. Accessed: 2024-08-03.

[33] MonkeyLearn. Sentiment analysis: Definition, examples, and applications. https://monkeylearn.com/sentiment-analysis/. Accessed: 2024-07-08.

[34] Mozilla Developer Network. What is a url? https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL, n.d. Accessed: 2024-07-26.

[35] Hamburg University of Technology. Data quality explored, n.d. Accessed: 2024-07-26.

[36] ResearchGate. Illustration of linear svm classifier separating the two classes. Image on ResearchGate, 2024. Accessed: 2024-08-05.

[37] ResearchMethod.net. Dissertation methodology, n.d. Accessed: 2024-07-17.

[38] Saed Sayad. Logistic regression, 2024. Accessed: 2024-08-03.

[39] Shehneela Naz Sayyida Tabinda Kokab, Sohail Asghar. Transformer-based deep learning models for the sentiment analysis of social media data. *science direct*, 14:1–12, 2022.

[40] Towards Data Science. Text pre-processing: Stop words removal using different libraries. *Towards Data Science*, 2019. Accessed: 2024-07-26.

[41] Mrs Snehal Shah1-Miss Akshata Bhat Miss Sumitra Singh Miss Arya Chavan-Master Aryan Singh. Sentiment analysis, 2024. Accessed: 2024-07-08.

[42] SmartDraw. What is a decision tree?, 2024. Accessed: 2024-08-03.

[43] Pure Storage. Machine learning performance metrics. https://www.purestorage.com/knowledge/machine-learning-performance-metrics.html, 2024. Accessed: 2024-09-01.

[44] Analytics Vidhya. Naive bayes explained. *Analytics Vidhya*, 2017. Accessed: 2024-08-05.

[45] Wikipedia. Support vector machine maximal margin hyperplane. Image on Wikipedia, 2024. Accessed: 2024-08-05.

[46] Wikipedia contributors. Lemma (morphology), 2024. Accessed: 2024-08-03.

[47] Wikipedia contributors. Random forest, 2024. Accessed: 2024-08-03.

[48] Shan Zhong Xiaokang Gong, Wenhao Ying and Shengrong Gong. Text sentiment analysis based on transformer and augmentation. *Frontiers in Psychology*, 13:1–9, 2022.