J. Electrical Systems 20-10s (2024):5322-5333

[1]Nisha Tayal
[2]Makhan Singh*
[3] Mallika Raj

A BERT-Based Technique on IMDb For False Movie Review Detection

Journal of Electrical Systems

Abstract: - Social media's popularity has grown alarmingly over the past ten years. In comparison to prior times, everyone is using technology more frequently. Many businesses use data from multiple websites to generate relevant information that can then be used for business objectives. Nowadays, people blindly trust the reviews available online and form a perception of any movie even before watching it. On websites like Amazon, IMDb, and Rotten Tomatoes, there is a wealth of textual information about movies. IMDb movie reviews are used to forecast user ratings for the films. Researchers in the field of machine learning have investigated a number of approaches to implementing the procedure with the highest level of accuracy. This paper focuses on finding false IMDb movie reviews by using a deep learning technique called BERT. In the proposed work, the BERT-base-uncased type is employed, which utilized pandas, torch, and transformer that produced an accuracy of 93% on the IMDb dataset.

Keywords: Machine learning, BERT, IMDb, Torch, Pandas, Transformer, XGBoost.

I. INTRODUCTION

Movies are the most convenient form of entertainment for most people. However, very few films are commercially successful and receive high reviews. There are numerous rating services that can assist movie enthusiasts in deciding which films to view and which to skip. Numerous websites offer movie reviews that reveal user opinions. The most popular of those include sites like IMDB and Rotten Tomatoes. The Internet Movie Database (IMDb) website is one example. IMDb is a website dedicated to movies and film production. Its material is highly comprehensive, including the actors and actresses that appeared in the film, a synopsis of the story, links to movie trailers, release dates for the film in various countries, and user reviews. The opinions of others and movie reviews typically have an impact on someone's decision to purchase or watch a film.

Individuals choices are influenced by their past experiences, personal beliefs, and the guidance they receive from fellow humans. Whenever someone intends to buy a new item or utilize a service, they often seek advice from others. Social media platforms frequently feature a continuous stream of comments, encompassing both favorable and unfavorable feedback, on a wide array of subjects, including products, services, businesses, institutions, economics, politics, entertainment, sports, and many others.

Machine learning utilizes a logarithmic approach to analyze words and gauge their sentiment, ranging from positive to negative. Through exposure to text samples with emotional content, machine learning algorithms autonomously acquire the ability to discern sentiment without human intervention. The study employed a sentiment analysis model built on machine learning to assess how well it performed on the IMDB review dataset. This research involves categorizing movie reviews based on the sentiment they convey and determining whether the sentiment is positive or negative. Given the frequent use of emotionally charged words in this research, a method has emerged for elucidating a film's overall sentiment using these linguistic cues.

The paper is structured as follows: Section 1 introduces the IMDB movie review. Section 2 provides an overview of "Related Work," discussing research closely related to the current study. Section 3 presents the methodology's details for practical implementation. Section 4 reveals the results. Finally, Section 5 concludes the paper.

II. RELATED WORK

A number of methodologies and complex algorithms are utilized to train machines to perform sentiment analysis. Each has its advantages and disadvantages. However, they can provide astonishing results when utilized in

[1] Electrical and Electronics Engineering, UIET, Panjab University, Chandigarh, India. nisha.tayal@pu.ac.in

[2] Computer Science and Engineering, UIET, Panjab University, Chandigarh, India. singhmakhan@pu.ac.in

[3] Computer Science and Engineering, UIET, Panjab University, Chandigarh, India. mallika.cyber@gmail.com

*Corresponding author: Makhan Singh

concert. Many studies on the sentiment analysis of text based on movie reviews have been conducted in recent years. Various methodologies were employed for sentiment analysis.

Tirath Prasad Sahu [1] used sentiment analysis to categorize the polarity of evaluations on a scale of 0 (strongly hated) to 4 (extremely praised). They extracted and ranked features before training a multilabel classifier to correctly label the reviews. Using SentiWordNet's lexical approach, they concentrated on two major areas: 1) feature selection and ranking, and 2) classification via machine learning approaches. They used the Rotten Tomatoes movie review dataset, which contained 8,000 polarized reviews, to evaluate six classifiers: Random Forest, Decision Tree, COCR, Bagging, KNN, and Naive Bayes. They discovered that the Random Forest classifier achieved the greatest accuracy of 88.95%.

Palak Baid [2] assessed movie reviews using Naïve Bayes, KNN, and Random Forest approaches. This study used these techniques to determine the polarity of tweets. Data was gathered from 2000 user-created movie reviews stored on the IMDb (Internet Movie Database) website. They processed 1000 favorable and 1000 negative reviews. This information was then translated to ARF format. The data was saved in the txt_token file, which was subdivided into two folders: positive and negative. The converted data was loaded into the WEKA tool using a text directory loader. The content was then pre-processed using the WEKA software. WEKA is an unrestricted freely downloadable application distributed under the (GNU) General Public License. It includes data-processing components. WEKA can do data mining tasks such as information pre-processing, binning, clustering, regression, and feature selection. The three classification algorithms employed are Naive Bayes, k-nearest neighbors, and Random Forest. Out of which, the best accuracy was given by Naïve Bayes with an accuracy of 81%.

Gurshobit Singh Brar [3] investigated sentiment analysis in movie reviews using feature-based opinion mining and supervised machine learning. The study sought to detect the polarity of reviews by using opinion words such as nouns, verbs, and adjectives. The reviews were divided into favorable and negative categories. The study used data from the Open Movie Database and the Natural Language Processing Toolkit for part-of-speech tagging. The method was tested on over 50 movie titles, each with up to 10 reviews, for a total of 500 reviews. The methodology included part-of-speech tagging, feature and opinion word extraction, and sentence and review polarity classification. The proposed approach classified movie reviews with an average accuracy of 81.22%.

H. M. Keerthi Kumar [4] developed a method that integrates machine learning features (such as term frequency (TF) and term frequency-inverse document frequency (TF-IDF)) with lexicon features (including positive-negative word count and connotation) to enhance accuracy and reduce complexity in sentiment classification compared to traditional classifiers like SVM, Naive Bayes, KNN, and Maximum Entropy. The hybrid approach aims to improve the distinction between positive and negative reviews by effectively capturing the context of the movie reviews. The process involves two parallel stages of feature extraction: one based on machine learning techniques and another based on lexicon analysis. Supervised learning algorithms Naive Bayes, Maximum Entropy, SVM, and KNN were applied to this combined feature set. Among these, Maximum Entropy proved to be the most accurate, achieving an accuracy rate of 83.93%. This approach demonstrates that integrating diverse feature types can significantly enhance classification performance in sentiment analysis. Nhamo Mtetwa [5] investigated how different feature extraction strategies affect supervised machine learning classifiers using a publicly available movie review dataset. The study's goal was to compare several classification algorithms and feature extraction approaches to find the most successful combination for movie review classification. The study compared three commonly used machine learning algorithms multinomial Naive Bayes, random forest, and support vector machine (SVM) using four performance metrics: accuracy, precision, recall, and F1-score. The study examined the results of various feature extraction methods and chose the top-performing combinations. The results showed that TF-IDF with SVM and bi-grams with multinomial Naive Bayes produced the greatest accuracy, both at 88%. The study emphasizes the necessity of using the right feature extraction techniques and classifiers to maximize performance in sentiment examination jobs.

Sumesh Kumar Nair [6] analysed sentiment on the IMDB dataset using a recurrent neural network, specifically Long Short-Term Memory (LSTM) networks, in conjunction with other techniques. LSTMs, a specialized type of RNN, were used to categorise movie reviews as negative or favorable. To train and evaluate the deep learning model, features were extracted using the 'word vector' method. To improve generalization and accuracy, the model architecture includes many hidden layers and includes technologies like Parametric Rectified Linear Unit

(PReLU), Dropout, and Normalization. The study also investigated the effects of different hyperparameters and neural network setups. The final model had an accuracy of 88.89 percent.

Nehal Mohamed Ali [7] conducted a study on sentiment analysis using DL networks, comparing various models to determine their effectiveness. The research used an IMDB dataset containing 50,000 movie reviews, which were evenly divided into positive and negative categories. The models tested included a multilayer perceptron (MLP), long short-term memory (LSTM) recurrent neural networks, convolutional neural networks (CNN), and a hybrid CNN-LSTM model. Data preprocessing involved using Word2Vec for word embedding. Among the models, the hybrid CNN-LSTM model achieved the highest accuracy at 89.2%. This model outperformed the CNN, which had an accuracy of 87.7%, and the MLP and LSTM models, which achieved accuracies of 86.74% and 86.64%, respectively. Additionally, the DL models demonstrated superior performance compared to traditional methods such as support vector machines (SVM), naive bayes, and recursive neural tensor networks (RNTN). The findings highlight the effectiveness of advanced DL models for sentiment analysis tasks.

Sumedh Shah [8] investigated the polarity of films based on reviews using machine learning algorithms, employing the IMDB movie review dataset for both training and testing. The study specifically tackled real-world issues such as class imbalance and train-test splits. Class imbalance, a common challenge in predictive applications like cancer discovery and fraud detection, was a focal point of the research. To address this issue, the study used various feature extraction methods, including Bag of Words and Term Frequency-Inverse Document Frequency (tf-idf), to process and analyze the reviews. The performance of the sentiment classification was assessed using support vector machines (SVM) and logistic regression classifiers. The research also involved computing the confusion matrix to demonstrate how class imbalance affects the accuracy of the models. The findings indicated that SVM combined with tf-idf achieved the highest accuracy at 87.07%, surpassing the performance of logistic regression. This highlights the effectiveness of SVM with tf-idf in handling class imbalances and accurately classifying movie review sentiments.

Zeeshan Shaukat [9] carried out research on extracting opinions from movie reviews using neural networks, specifically trained on Stanford's "Movie Review Database." The goal was to provide quantitative assessments of films' favorability or unfavourability, thereby offering qualitative insights into various aspects of the films. This was achieved using an enhanced Bag of Words (BoW) mechanism for sentiment analysis. In their methodology, the research team focused on text sentiment analysis, which involved classifying reviews as either positive or negative. This classification was facilitated by a dataset that divided the reviews into these two categories. The lexicon-based approach was employed to calculate the semantic orientation of words within the reviews, which contributed to the sentiment analysis. The enhanced BoW mechanism was a critical aspect of the study, as it significantly improved the accuracy of the sentiment classification. The research achieved a training accuracy of 91% and a validation accuracy of 86%, demonstrating the effectiveness of their approach in accurately analyzing and classifying movie reviews based on sentiment.

Reza Maulana [10] performed sentiment analysis on movie reviews using a support vector machine (SVM) with information gain for feature selection. SVM is a popular text classification system; however, it has limitations in feature selection. Information gain improves this by allowing for faster and more consistent convergence. The approach was evaluated using the Cornell and Stanford movie review datasets. On the Cornell dataset, SVM achieved 83.05% accuracy, with information gain reaching 85.65%. On the Stanford dataset, SVM alone scored 86.46% accuracy, whereas SVM with information gain improved to 86.62%. This demonstrates that SVM with knowledge gain is more accurate for film review sentiment analysis.

Arafat Habib Quraishi [11] investigated the effectiveness of four ML methods for sentiment analysis using the IMDB review dataset. Two of the methods used neural networks (LSTM and GRU), whereas the other two did not. GRU outperformed LSTM among neural network techniques, and SVM outperformed multinomial naive bayes among non-neural networks. GRU accomplished the most noteworthy precision of 89.0%, making it the best of the four calculations.

Saeed Mian Qaisar [12] employed the Long Short-Term Memory (LSTM) classifier in conjunction with the Recurrent Neural Network (RNN) method to investigate feelings in IMDb movie reviews. The LSTM was combined with the Adam optimizer, an adaptive learning rate optimization approach developed specifically for deep neural networks. To calculate different learning rates for each parameter, the algorithms employed adaptive learning rate techniques. The classification results were evaluated based on accuracy, with the greatest classification accuracy being 89.9%.

Samuel Onalaja [13] conducted the research, comparing classification models for determining aspect-based text sentiment and predicting binary sentiments in film reviews based on genre and aspect-specific variables. Some of Techniques like Logistic Regression, Naive Bayes, Support Vector Machine, and Recurrent Neural Network Long-Short-Term Memory were all tested for complete categorization. Sentences were divided depending on film elements obtained by aggregating aspect terms through lexicon-based, supervised, and unsupervised learning. Driving variables were randomly allocated to numerous movie characteristics, and the effect of each component and genre on sentiment classification was thoroughly investigated. The study discovered that sentiment prediction models performed better when specific features and genres were given more driving variables. The study obtained an accuracy of 87.3% by analyzing 3,000 movie reviews, evenly split between positive and negative ratings.

Joao Ramos [14] research investigates the use of sentiment features to anticipate movie evaluation scores, employing three major SA approaches: lexicon-based, supervised machine learning, and hybrid. They employed IMDb movie reviews, with TF-IDF and Doc2Vec features producing 87% results. The hybrid technique produced improved results but did not differ much from the SML approaches.

Sichang Su [15] research looks at the sentimental trends of movie reviews from two films in the same series. The Term Frequency-Inverse Document Frequency (TF-IDF) algorithm determines the significance of words in reviews, while the Text Blob sentiment analysis library assigns sentiment scores. An SVM model identifies the reviews with an accuracy of 85.2%. This study assists the viewer in making movie choices. The summary of literature survey is shown in Table 1.

Table 1. Literature Survey

Ref No	Name	Approach	Model	Dataset	Accuracy
[1]	T. P. Sahu et al. (2016)	Machine Learning techniques.	RF, DT, Bagging, KNN and NB	8000 IMDb movie reviews.	88.95%.
[2]	Palak Baid et al. (2017)	ML techniques.	NB, K-NN and RF	2000 created reviews from IMDb	81%.
[3]	G. Singh Brar et al. (2018)	Natural Language Processing, Sentiment Lexicon	Support Vector Machine, Lexicon and Naïve Bayes	500 IMDb Movie reviews.	81.22%
[4]	H. M. K. Kumar et al. (2018)	ML features (TF, TF-IDF) with Lexicon features	Nave Bayes, Maximum Entropy, SVM, and KNN	5000 IMDb movie reviews.	83.93%
[5]	N. Mtetwa et al. (2018)	ML classifiers.	Multinomial Naive Bayes, Random Forest, and SVM	50k Stanford University's ACL IMDB movie reviews.	88%
[6]	S. K. Nair et al. (2018)	Deep Learning	LSTM Recurrent Neural Networks.	50k IMDb movie reviews	88.89%.
[7]	N. Mohd Ali et al. (2019)	Deep Learning	CNN_LSTM	50k IMDb movie reviews	89.2%
[8]	S. Shah et al. (2019)	ML algorithms	SVM and logistic regression.	50k IMDB reviews.	87.07%
[9]	Z. Shaukat et al. (2020)	Lexicon, Deep learning.	Multilayer perceptron, Lexicon, NN	Stanford's Movie Review Database.	86%
[10]	Reza Maulana et al. (2020)	ML	Support Vector Machine	Cornell and Stanford datasets.	86.62%.
[11]	A H Quraishi et al. (2020)	Machine Learning Algorithms.	LSTM and GRU, MNB, and SVM.	IMDB review dataset	89.0%.
[12]	S. M. Qaisar et al. (2020)	Deep Learning.	LSTM and RNN	IMDb movie reviews.	89.9%.
[13]	S. Onalaja et al. (2021)	ML Algorithms and DL	LR, NB, SVM, and RNN LSTM	3,000 movie reviews.	87.3%
[14]	Joao Ramos et al. (2022)	lexicon-based machine learning.	Expanded OntoSenticNet EON	IMDB movie review (1 million reviews)	87%
[15]	Sichang Su (2022)	Machine Learning	SVM	3k IMDb reviews	85.2%

III. METHODOLOGY

In the proposed framework, sentiment analysis is employed on IMDb Movie Reviews. The suggested model's input is a collection of reviews whose polarity must be determined. The output consists of reviews, each of which has a different polarity. The steps involved in sentiment analysis are as follows: preprocessing the dataset, feature Extraction, feature selection, and finally classification using different classifiers. The suggested model's overall workflow depicts in Figure 1.

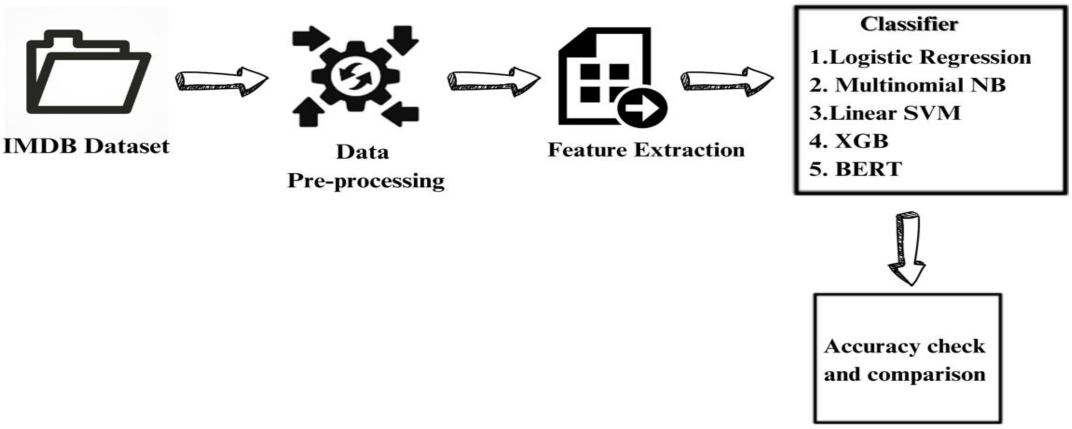

Figure 1. Methodology of the proposed work

A. Data Collection

1) Datasets: IMDb is the most popular resource for obtaining film information globally. The IMDb Movie Review Dataset is included in the proposal due to its popularity and the availability of a large range of reviews regarding a particular film. It is one of the basic metrics used in film reviews for SA. This dataset contains 50,000 sentimentally rated (positive or negative) movie reviews from IMDB. There are exactly 25,000 positive and negative reviews. After importing a CSV file into Python, the proposed study trained the supervised learning algorithms using various train-test splits. In each of the reviews, users are able to give the film a score between 1 and 10. The proposed study defines a review as negative if it receives a rating of less than 4 and positive if it receives a rating of more than 7, in order to convert this rating scale to a binary label. Reviews that fall in between these two ranges are ignored.

2) Dataset Visualization: In this section, various parameters and factors influencing classification are identified using the Python environment, and the results are shown. To visualize textual data, the proposed study used the Python wordcloud library. It exhibits a catalog of words, where the significance of each word is conveyed through variations in either font size or color. This style is helpful for quickly identifying the most important words. A wordcloud of positive terms that stand out in positive evaluations, with their ranking varying with the size of the word in the wordcloud is shown in Figure 2.

A wordcloud of the negative terms that stand out in the negative reviews, with their ranking varying with the size of the word in the wordcloud is shown in Figure 2.

Figure 2. Positive wordcloud

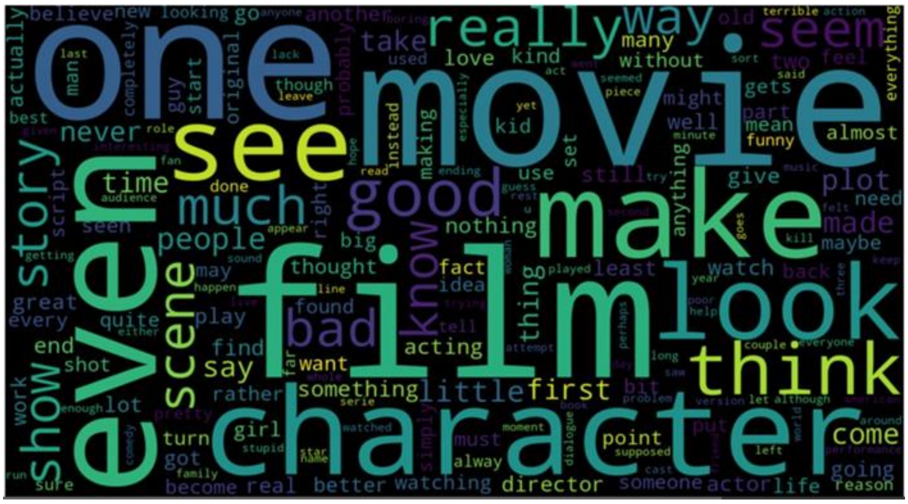

Figure 2. Negative wordcloud

B. Data Cleaning and Pre-Processing

Data cleaning is the process of repairing or deleting incorrect, corrupted, poorly formatted, duplicate, or incomplete data from a dataset. There are numerous ways for data to be duplicated or mislabeled when merging multiple information sources. In order to enhance the performance of the proposed model, various operations have been performed on the gathered data. This also eliminated extraneous noise from the data to aid the model's classification. To clean the data, the proposed study involves the following steps: removing HTML tags, stop words and special characters, square brackets, URLs, and the noisy text. It also utilized normalization to remove all punctuation and convert it into lowercase, and numerals were replaced with their corresponding word equivalents.

C. Feature Extraction

Feature extraction is the process of reducing the number of attributes. Unlike feature selection, which evaluates existing qualities based on their predictive relevance, feature extraction changes the attributes into features. The proposed study makes use of the labeled dataset that was acquired. The remaining unlabeled data will be utilized to assess the models. To classify pre-processed data, some machine learning and deep learning methods were applied. Some of the most prominent feature extraction approaches are: Bag-of-Words, Count Vectorizer, Word Frequency, and TF-IDF.

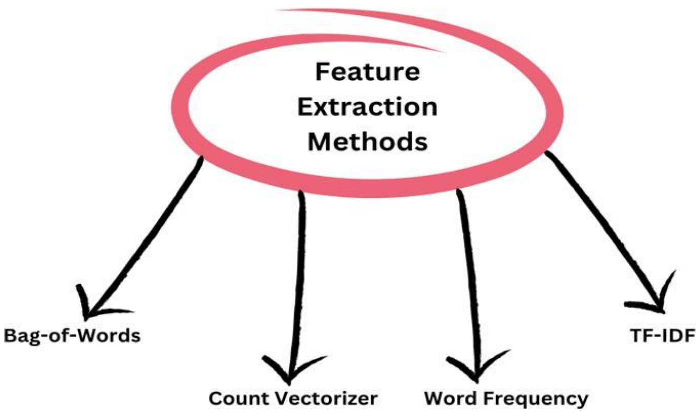

Figure 3. Feature Extraction Methods

- Bag-of-words: The bag-of-words (BOW) model is a representation that divides any text into fixed-length vectors based on the frequency of each word. Vectorization is a common name for this procedure. In document classification, the BoW model is utilized, and each word is used as a feature to train the classifier.
- Count Vectorizer: The procedure consists of tokenizing a set of documents from the datasets and then creating a set of vocabulary for those terms. As a consequence, the length and numerical value of each word are

established according to how frequently it appears in the vocabulary. Words that do not occur may contain zeros as their value and are referred to as sparse.

- Word Frequency: It measures how frequently a word appears in a corpus or text. It can be stated as a relative frequency, which is the proportion of occurrences to the total number of words, or as an absolute frequency, which is the simple count of occurrences.
- TF-IDF: Term Frequency-Inverse Document Frequency is a statistical measure for information retrieval. It is a technique for calculating term weights in a document. It represents the significance of a word in a document in comparison to the total corpus. The number of times a word appears in the text enhances its significance; however, this is balanced by the word frequency in the corpus (data-set). Hence, TF is defined as:

$$TF = \frac{No.of\ times\ word\ apperas\ in\ the\ document}{Total\ no.of\ terms\ in\ the\ document}$$

And IDF is defined as:

$$IDF = log \frac{No.of\ the\ documents\ in\ the\ corpus}{No.of\ documents\ in\ the\ corpus\ contain\ the\ term}$$

Therefore, TF-IDF is calculated by multiplying TF and IDF scores:

$$TF - IDF = TF * IDF$$

- N-grams: the last feature extraction is n-grams. N-grams are a collection of words that frequently appear inside certain windows; thus, when $n = 1$ it is Unigram, $n = 2$ it is bigram; $n = 3$ it is trigram and so on.

The fundamental idea behind n-grams is that they represent linguistic structure from a statistical perspective, such as what letter or word is most likely to come after the given one. One has more context to work with as n-gram length increases (i.e., as n increases).

Word sequences are provided as follows:

$$w_1^n = w_1 \dots w_n$$

Illustrations for N-gram probabilities are given by:

$$\begin{aligned} &\mathbf{P}(\mathbf{w_n} \mid \mathbf{w_1} \dots \mathbf{w_{n-1}}) \approx \mathbf{P}(\mathbf{w_n}) \ \text{unigram} \\ &\mathbf{P}(\mathbf{w_n} \mid \mathbf{w_1} \dots \mathbf{w_{n-1}}) \approx \mathbf{P}(\mathbf{w_n} \mid \mathbf{w_{n-1}}) \ \text{bigram} \\ &\mathbf{P}(\mathbf{w_n} \mid \mathbf{w_1} \dots \mathbf{w_{n-1}}) \approx \mathbf{P}(\mathbf{w_n} \mid \mathbf{w_{n-1}} \mathbf{w_{n-2}}) \ \text{trigram} \end{aligned}$$

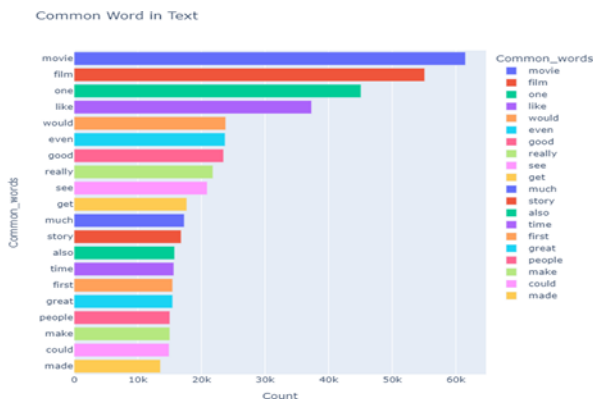

Figure 5. Unigram

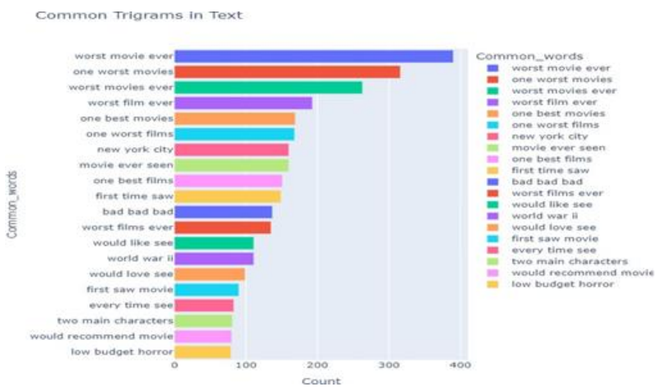

Figure 4. Trigram

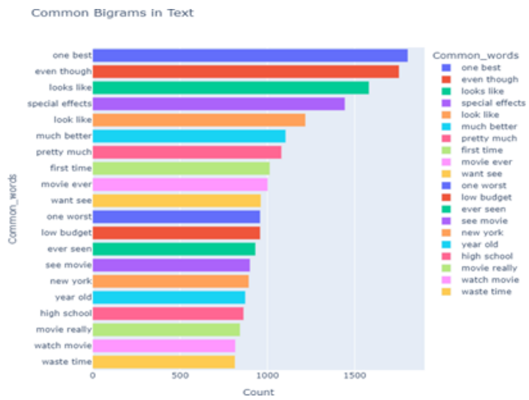

Figure 7. Bigram

D. Classification

Classification is a machine learning method in which the model attempts to predict the correct label for a given set of input data. In classification, the model is thoroughly trained using the training data and then assessed on test data before making predictions on fresh, unused data. The proposed study used five different machine learning algorithms on the IMDB review dataset. Below is a brief description of all the algorithms employed in this study.

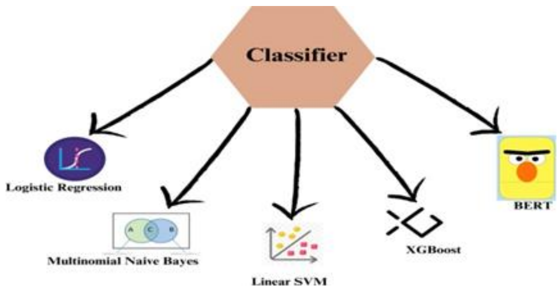

Figure 8. Classifiers

Logistic Regression: Logistic regression [8] is a classification algorithm that uses supervised learning to predict the likelihood of a target variable. It forecasts the outcome of a categorical dependent variable. As a result, the answer must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or false, etc., but rather than providing an exact value between 0 and 1, it provides probabilistic values that are in the range of 0 and 1. In logistic regression, the proposed study fits an S-shaped logistic function (Sigmoid function), which predicts two maximum values (0 or 1), rather than a regression line.

Logistic regression uses the sigmoid function:

$$Y = e^{(b0+b1*x)}/(1 + e^{(b0+b1*x)})$$

Here b0 is the bias and b1 is the coefficient for the values x and y.

Multinomial Naïve Bayes: The Multinomial Naive Bayes algorithm [18] is a version of the Naive Bayes algorithm used in machine learning. It is excellent for multinomial distributed datasets. It explains the likelihood of witnessing counts in a variety of categories, and hence it is best suited for features that represent counts or count rates, which fits our movie review use case. It is frequently used in text classification, where the features are connected to word counts or frequencies within the documents to be categorized. For a document or text d and class c, Naïve Bayes predicts the probability of class c for text d with the following conditional probability:

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)}$$

Linear SVM: The Linear Support Vector Machine (Linear SVC) [19] is an algorithm that seeks to find a hyperplane that maximizes the distance between categorized samples. The proposed study has used Linear SVM only when the data is perfectly linearly separable. Perfectly linearly separable means that the data points can be divided into two classes using a single straight line. The proposed study has a training dataset consisting of input feature vectors X and their corresponding class labels Y.

In the case of linear classifications, the equation is $wx + b = 0$.

The vector W indicates the hyperplane's normal vector, i.e. the direction perpendicular to the hyperplane. The parameter b in the equation denotes the hyperplane's offset or distance from the origin along the normal vector

w. The following formula is used to compute the distance between a data point x_i and the decision boundary:

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

Where ||w|| denotes the Euclidean norm of the weight vector w. Euclidean norm of the normal vector W. For a Linear SVM classifier:

$$\hat{y} = \begin{cases} 1 & : \ w^T x + b \geq 0 \\ 0 & : \ w^T x + b < 0 \end{cases}$$

XGBoost: is an optimized distributed gradient boosting library developed for efficient and scalable machine learning model training. XGBoost [20] is an abbreviation for "Extreme Gradient Boosting," and it has become one of the most popular and widely used machine learning algorithms due to its ability to handle large datasets and achieve cutting-edge performance in many machine learning tasks such as classification and regression. The objective function [16] (loss function and regularization) at iteration t needs to minimize the following:

Real value (label) known from the training data-set

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t)$$

Can be seen as f(x + Δx) where x = $\hat{y}_i^{(t-1)}$

BERT: The language model suggested by Google AI is called Bidirectional Encoder Representation from Transformers [21]. A BERT model's architecture is a multi-layer bidirectional transformer encoder that learns to predict the characteristic representation of the input sequence (contextualized embedding). The bidirectional architecture allows for the integration of information from both sides, enabling BERT to handle a variety of jobs. BERT is just transformer architecture with an encoder stack. The following diagram illustrates the BERT architecture [17].

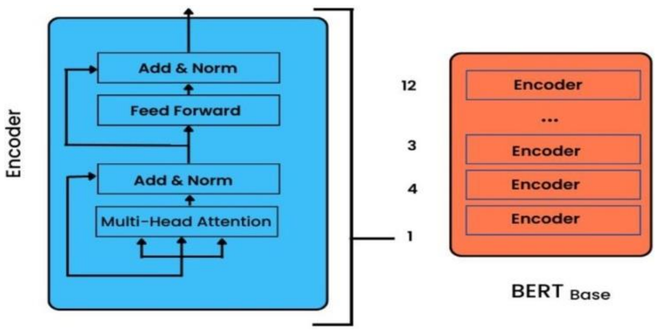

Figure 10. BERT Architecture [17]

The pre-trained BERT model utilized in this study is BERT-base-uncased. A [CLS] token (classification) and a [SEP] token (separator) are added to the review in order to make it suitable for BERT. Since BERT requires fixed-length sentences as inputs, a maximum length parameter is selected before training such that sentences longer than this length is clipped and those shorter are padded with [PAD] tokens. This parameter is defined by the quantiles of size observed after the process of tokenization. Furthermore, the attention mask (0 for [PAD] tokens and 1 for others) is obtained to allow the model to focus on non-padding tickets. The embedding of the [CLS] token is extracted into a linear layer for classification after passing through the BERT model, as this is where the contextualized representation of the movie review is stored.

Table 2. Comparison of various classification algorithms

Algorithm	Functions	Benefits	Drawbacks
Logistic	In the context of movie reviews,	It is trained quickly and is suitable	It struggles when there are

Regression:	this helps identify which words or phrases are most indicative of positive or negative sentiment.	for handling large datasets of movie reviews. It is designed for binary classification and works well in movie reviews, where it has two classes to predict.	intricate interactions between words or phrases in a movie review that contribute to sentiment. It doesn't handle variable-length sequences effectively, which is problematic when analyzing reviews of varying lengths.
Multinomial Naive Bayes:	It is employed to classify movie reviews into different sentiment categories, typically binary (positive or negative). It learns from a labeled dataset of movie reviews, where each review is associated with a sentiment label.	It can handle large datasets and high-dimensional feature spaces efficiently, making it suitable for processing large collections of movie reviews. It helps in understanding which words or phrases are most indicative of positive or negative sentiment in movie reviews.	Choosing the right features, such as which words to include or exclude, is a non-trivial task. This increases the computational complexity and memory requirements of MNB, especially when using TF-IDF representations.
Linear SVM:	It is generally robust to noisy or irrelevant features in the data, as it focuses on finding the optimal decision boundary rather than modeling complex relationships between features.	It is robust in handling high-dimensional data and performs well even when dealing with a large number of features. It reduces computational complexity and memory requirements, making it efficient for large-scale text analysis.	Noise in the feature space negatively impacts model performance. Training time is limited in this case.
XGBoost:	It builds an ensemble of decision trees, which are often shallow trees, and combines their predictions to make a final classification decision. This ensemble approach helps improve predictive accuracy and robustness.	It provides options for handling imbalanced data, such as using weighted loss functions or subsampling, which leads to better performance on minority classes. Its flexibility allows incorporating different types of information into sentiment analysis model, making it adaptable to diverse datasets.	It treats words as independent features and doesn't capture complex linguistic relationships, which limits its ability to analyze sentiment in the context of movie reviews effectively. It is sensitive to noisy or irrelevant features in the data, which affect model performance.
BERT:	BERT is fine-tuned for sentiment classification tasks, where the goal is to predict the sentiment (positive or negative) of movie reviews. The model takes the entire review as input and outputs the predicted sentiment label.	It captures the contextual meaning of words and phrases in movie reviews, allowing it to understand nuances, sarcasm, and complex sentence structures, which is crucial for accurately analyzing sentiment in reviews. It is inherently robust to noisy or irrelevant features in text data, which allows it to focus on the most informative aspects of the reviews.	Fine-tuning BERT model for specific tasks takes a significant amount of time, especially to experiment with different hyper parameters or perform cross-validation. Training BERT from scratch is time-consuming and resource-intensive, especially with limited access to high-performance computing resources.

IV. RESULT

The effectiveness of the recommended model is estimated by comparing it with other models using the IMDb dataset. The performance of the suggested sentiment analysis model is measured using several classifiers. For all performance measures, the BERT classifier outperforms other classifiers in the performance evaluation. BERT outperforms logistic regression, multinomial naive bayes, linear SVM, and XGBoost in accuracy by 93%. The proposed research has evaluated the results of the classification models based on accuracy. Accuracy can simply be defined as the ratio of accurately predicted observations to the total number of observations. The proposed study claims that the better the model, the higher the accuracy.

As a result, accuracy is determined by the following equation:

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

Where TP = True Positive, TN = True Negative, FP = False Positive, and FN = False Negative.

Table 3. Accuracy of various classification algorithms

Classifier	Accuracy
Logistic Regression	87%
Multinomial Naive Bayes	85%
Linear SVM	88%

XGBoost	83%
BERT (Proposed)	93%

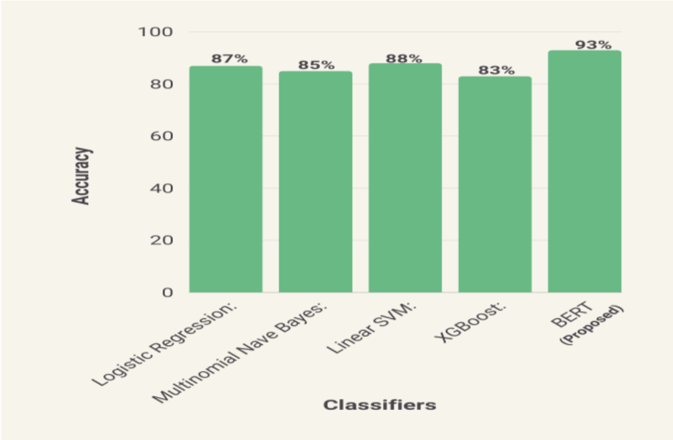

Figure 5. Accuracy Graph

The above graph shown in figure 8 represents the proposed model evaluation among the logistic regression, multinomial naive bayes, linear support vector machine, XGBOOST, and BERT models. The proposed work stated that the Bert model has outperformed the rest of the other models with a peak accuracy of 93%, which is significantly higher and shows effective results in detecting sentiment analysis of movie reviews on various platforms.

V. CONCLUSION AND FUTURE WORK

Sentiment analysis, often known as opinion mining, is the technique of extracting and categorizing sentiments into positive, negative, or neutral categories from text data. This research focuses mainly on a comparison of various machine learning techniques for extracting sentiments from text. They are less complicated and more effective. The proposed work uses BERT and other machine learning algorithms to classify the IMDB movie review dataset. Importantly, implementing BERT has proven to be significantly less difficult than traditional methods. In this research, the BERT classifier is combined with the AdamW optimizer to automatically classify IMDb movie reviews. BERT with the highest accuracy can be used as a standard for all other algorithms. It gives enough information to help enhance predictions in future study efforts. It can be suggested that the cleaner the data, the better the effectiveness of an algorithm in predicting the success rate of movies.

Although sentiment analysis has advanced, current techniques are still sensitive to poorly organized and sarcastic messages, which emphasize the need for more trustworthy language models. Researchers can investigate the use of reinforcement learning approaches to train models that can handle confusing and toxic texts. Other feature selection techniques, such as particle swarm optimization (PSO), genetic algorithm (GA), Chi-Square, and others, can be used in future studies to compare optimal outcomes. Currently, English text is mostly used for training sentiment analysis models. However, there is a rising demand for sentiment analysis algorithms that can operate in several languages. A larger audience would be able to understand the sentiment expressed in several languages with the use of cross-lingual sentiment analysis. Researchers can investigate the use of transfer learning approaches to create sentiment analysis models that are multilingual.

REFERENCES

[1] T. P. Sahu and S. Ahuja, "Sentiment analysis of movie reviews: A study on feature selection & classification algorithms," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, India, 2016, pp. 1-6, 2016.

[2] Baid, Palak & Gupta, Apoorva & Chaplot, Neelam. “Sentiment Analysis of Movie Reviews using Machine Learning Techniques”, International Journal of Computer Applications. 179. 45-49. 10.5120/ijca2017916005. 2017.

[3] Gurshobit Singh Brar, Asst. Prof. Ankit Sharma, “Sentiment Analysis of Movie Review Using Supervised Machine Learning Techniques”, International Journal of Applied Engineering Research ISSN, 2018.

[4] Keerthi Kumar, H M & Harish, B S & Darshan, H. “Sentiment Analysis on IMDb Movie Reviews Using Hybrid Feature Extraction Method”, International Journal of Interactive Multimedia and Artificial Intelligence. InPress. 1.10.9781/ijimai.2018.12.005, 2018.

[5] N. Mtetwa, A. O. Awukam and M. Yousefi, "Feature Extraction and Classification of Movie Reviews," 5th International Conference on Soft Computing & Machine Intelligence (ISCMI), Nairobi, Kenya, 2018, pp. 67-71, doi: 10.1109/ISCMI.2018.8703235, 2018.

[6] Sumesh Kumar Nair, Ravindra Soni, “Sentiment Analysis on Movie Reviews using Recurrent Neural Network”, ICONIC RESEARCH AND ENGINEERING JOURNALS, e-ISSN: 2456-8880, pp. 242-251, volume 1 issue April 2018.

[7] Ali, Nehal Mohamed and Abd El Hamid, Marwa Mostafa and Youssif, Aliaa, “Sentiment Analysis for Movies Reviews Dataset Using Deep Learning Models”, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.9, No.2/3, May 2019.

[8] Shah, Sumedh & Anuse, Alwin & Kute, Rupali, “Classification of Sentiment Based on Movie Feedback Given by Audiences”, International Journal of Recent Technology and Engineering (IJRTE). 8. 2594-2602. 10.35940/ijrte. D7237.118419, 2019.

[9] Shaukat, Zeeshan & Zulfiqar, Abdul Ahad & Xiao, Chuangbai & Azeem, Muhammad & Mahmood, Tariq, “Sentiment analysis on IMDB using lexicon and neural networks”, SN Applied Sciences. 2. 10.1007/s42452-019-1926-x, 2020.

[10] Maulana, Reza & Rahayuningsih, Panny & Irmayani, Windi & Saputra, Dedi & Jayanti, Wanty, “Improved Accuracy of Sentiment Analysis Movie Review Using Support Vector Machine Based Information Gain”, Journal of Physics: Conference Series. 1641. 012060. 10.1088/1742-6596/1641/1/012060, 2020

[11] Quraishi, Arafat, “Performance Analysis of Machine Learning Algorithms for Movie Review”, International Journal of Computer Applications. 177. 7-10. 10.5120/ijca2020919839.,2020.

[12] S. M. Qaisar, “Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory,” 2020 2nd International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, pp. 1-4, doi: 10.1109/ICCIS49240.2020.9257657, 2020.

[13] Samuel Onalaja, Eric Romero, Bo Yun, Faizan Javed, “Aspect-based Sentiment Analysis of Movie Reviews”, SMU Data Science Review: Vol. 5: No. 3, Article 10, 2021.

[14] Joao Ramos, Diogo Apostolo, Hugo Goncalo Oliveira, “Movie rating prediction using sentiment features”,Language Resources and Evaluation Conference (LREC 2022), Marseille, 24 June 2022.

[15] Sichang Su, “Sentimental Analysis Applied on Movie Reviews”, Journal of Education, Humanities and Social Sciences ADEIM vol.3, 2022.

[16] XGBoost Available: https://dimleve.medium.com/xgboost-mathematics-explained-58262530904a, [Accessed: 21-Feb-2024].

[17] BERT Architecture [Online] Available: https://vitalflux.com/bert-vs-gpt-differences-real-life-examples/, [Accessed: 22-Feb-2024].

[18] Ravi Kumar J, Angeline Benitta, “IMDB Movie Reviews Sentiment Classification Using Deep Learning”, Easy Chair No 7529, march 11,2022.

[19] Support Vector Machine (SVM)[Online]Available: https://www.geeksforgeeks.org/support-vector-machine-algorithm/, [Accessed: 3-Jun-2024].

[20] XGBoost [Online] Available:https://www.spiceworks.com/tech/artificial-intelligence/articles/xgboost-vs-random-forest-vs-gradient-boosting, [Accessed: 8-Jun-2024].

[21] BERT [Online] Available: https://www.analyticsvidhya.com/blog/2021/06/why-and-how-to-use-bert-for-nlp-text-classification/, [Accessed: 12-Jun-2024].