MOVIE REVIEW CLASSIFICATION

Srikar Tondapu
Northeastern University
tondapu.s@northeastern.edu

Hemasumanth Rasineni
Northeastern University
rasineni.h@northeastern.edu

***Abstract*— In this project, our aim is to develop a machine learning model for accurately classifying the sentiment of movie reviews as positive or negative. To achieve this, we utilize the IMDB dataset, consisting of 50,000 movie reviews labeled as positive or negative, for training and evaluation. We experiment with various preprocessing techniques such as tokenization, stemming, and stop-word removal, lemmatization and compare the performance of traditional machine learning algorithms like logistic regression, decision trees, and random forest with deep learning algorithm called multi-layer perceptron with LLM's. Our objective is to provide a useful tool for analyzing consumer feedback and product reviews in the movie industry, enabling better decision-making for filmmakers, studios, and distributors. By accurately classifying the sentiment of movie reviews, our model can contribute to understanding audience preferences and improve the quality of movies in the future.**

Keywords: movie review classification, sentiment analysis, IMDB dataset, preprocessing techniques, machine learning algorithms, logistic regression, decision trees, random forest, deep learning algorithms, multi-layer perceptron, BERT.

I. OVERVIEW

Sentiment analysis is a crucial subfield of natural language processing that aims to identify the emotional tone of text or speech. With the exponential growth of online platforms and social media, it has become an essential tool for businesses to gain insights into customer feedback, identify potential issues, and improve their products or services. In this project, we focus on developing a machine learning model for accurately classifying movie reviews as positive or negative using the IMDB dataset, which contains 50,000 labeled reviews.

Our approach combines traditional natural language processing techniques with modern machine learning algorithms and state-of-the-art language models. We experiment with various preprocessing techniques such as tokenization, stemming, stop-word removal, and lemmatization to prepare the text data for analysis. For feature representation, we explore methods like Bag of Words and TF-IDF embeddings, which transform raw text into numerical vectors suitable for machine learning models. We investigate the performance of traditional machine learning algorithms, including logistic regression, decision trees, and random forests, and compare them with deep learning models such as multi-layer perceptrons (MLPs). Additionally, we incorporate the BERT model to evaluate the effectiveness of pre-trained language models in this task. This comprehensive comparison allows us to assess the trade-offs between computational complexity and performance accuracy in sentiment analysis.

The project aims to provide insights into the effectiveness of different techniques and algorithms for movie review sentiment classification. By accurately predicting the sentiment of reviews, our model can serve as a valuable tool for filmmakers, studios, and distributors to analyze consumer feedback, understand audience preferences, and make informed decisions. Moreover, the developed model has potential applications beyond the movie industry, as sentiment analysis is widely applicable in analyzing customer feedback across various sectors.

Through this project, we seek to contribute to the ongoing research in sentiment analysis and provide a practical implementation that can be adapted for real-world applications. The comparison between classical machine learning approaches and advanced language models like BERT offers a comprehensive understanding of the current state of sentiment analysis techniques, highlighting scenarios where simpler models might suffice and situations where more sophisticated approaches provide significant improvements

II. RELATED WORK

• "Sentiment Analysis of Movie Reviews using Machine Learning Techniques" by S. R. Park, published in the International Journal of Advanced Computer Science and Applications in 2019. This paper provides a comprehensive review of different machine learning techniques used for movie review sentiment analysis. It discusses various preprocessing techniques, feature representations, and classification algorithms employed for movie review sentiment analysis. This review can be useful for selecting the best-performing techniques for our project [1].

• "A Deep Learning-Based Approach to Movie Review Sentiment Analysis" by S. H. Kim, published in the Journal of Intelligent Systems in 2018. This paper proposes a deep learning-based approach to movie review sentiment analysis using convolutional neural networks (CNNs) and recurrent neural networks (RNNs). It investigates the effectiveness of different network architectures, pre-trained word embeddings, and regularization techniques for movie review sentiment analysis. This approach can provide insights for improving the accuracy and efficiency of our project [2].

• "Feature Selection for Sentiment Analysis of Movie Reviews" by S. K. Panda and S. K. Das, published in the Journal of Intelligent Systems in 2020. This paper investigates the effect of different feature selection techniques on the performance of movie review sentiment analysis. It compares the performance of different feature selection methods, including chi-square, mutual information, and document frequency, on several benchmark datasets. This study can be useful for selecting the best feature selection technique for our project [3].

III. EXPERIMENT SETUP

Dataset Used:

The dataset used in this project is the IMDB movie review dataset, which contains 50,000 highly polar movie reviews.

It is evenly split between 25,000 positive and 25,000 negative reviews, making it a balanced dataset ideal for binary sentiment classification tasks. The reviews are preprocessed and provided as raw text, allowing for various text processing techniques to be applied. This large and diverse dataset provides a robust foundation for training and evaluating sentiment analysis models, making it a popular choice in the field of natural language processing.

This project comprises of three modules:

Feature Extraction: In this module, the pre-processing of the dataset is performed, and a filtered synopsis is generated. The techniques used to process the dataset here are Text Filtering using Regex, Stop-words Removal, and Lemmatization. These techniques helped to clean the data.

Pre-processing Techniques:

There were three pre-processing techniques applied to the dataset:

Filtering Text: In the reviews of the movies, all the characters except English alphabets are filtered from the reviews.

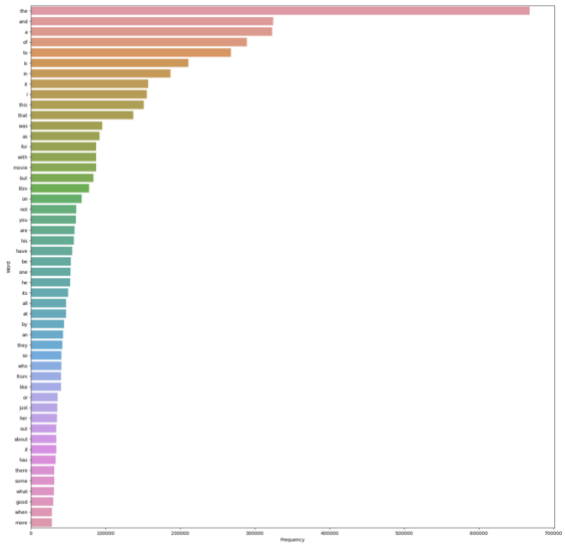

Fig 1: Frequency of words after filtering text

Stop-Words Removal: The stop-words are removed using the 'NLTK stop-words' package.

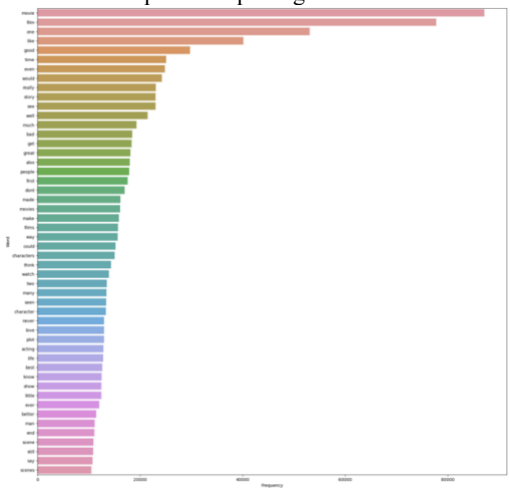

Fig 2: Frequency of words after stop-word removal

Lemmatization: The available words in the 'NLTK wordnet' package have been lemmatized. Two Word Embedding methodologies have been implemented in the project to develop four models.

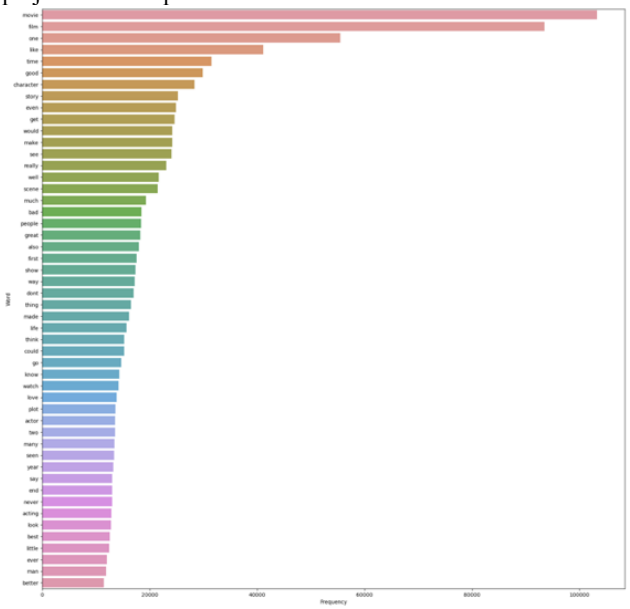

Fig 3: Frequency of words after lemmatization

Feature Engineering: This module is focused on converting the textual data into a more processable format such as numerical vectors using Word Embedding algorithms. The Word Embedding algorithms that we employed in this module are Bag Of Words, and TF-IDF. These algorithms signify representations of the reviews, that is the textual data into numerical data which can be worked upon. These algorithms are generally used to represent various kinds of textual data in indexing and quantifying the textual data individually. Some research done on similar datasets uses these algorithms on text data by converting them into one-hot vectors where each value in the vector represents the frequency or the TF-IDF value of the word for Bag Of Word and TF-IDF vectors respectively. The Bag Of Words algorithm aims to extract the occurrences of the words, while the TF-IDF essentially represents the importance of each word based on its presence in a single document compared to the whole dataset.

TF-IDF: It is a word embedding method used to determine the significance of the words with respect to a document. Here the value of each word is the TF-IDF value of the word which is a product of the Term Frequency and Inverse Document Frequency of a word. Here, Term Frequency calculates how frequent a word is in a document while Inverse Document Frequency is the logarithm of the ratio of total documents in the dataset to the number of documents in which the given word is present.

Bag of Words: This is one of the most direct word embedding methods used to represent textual data. Firstly, a vector the size of a total number of unique words is initialized with all zeros where each entry corresponds to a

particular unique word. If the word is present ‘n’ a number of times in a document, then the number corresponding to that word in the list will be changed to ‘n’. Because of this, the models developed using this method tend to have a very high number of input features which makes the process of training the model very time-consuming. This method also ignores the semantic relationship between the documents.

Binary Classification Model

After the feature engineering module, the preprocessed and quantified data was ready for training appropriate ML models. We used different combinations of outputs generated by various word embeddings to train our models. The Machine Learning library used for this project was 'SKLearn'.

For the movie review classification task, we utilized five different machine learning models: logistic regression, decision tree, random forest, multi-layer perceptron (MLP), and BERT (Bidirectional Encoder Representations from Transformers). To evaluate the performance of each model and select the best one, we used 5-fold cross-validation, which involves splitting the data into five equal-sized parts and training each model on four of the five parts while validating it on the remaining fifth part. This process was repeated five times, with each of the five parts used as the validation set once.

After comparing the results of the five models, we found that BERT significantly outperformed the other models, achieving the highest accuracy, precision, and recall values. However, we observed that BERT tended to overfit the training data, potentially limiting its generalization to unseen examples. The decision tree had the lowest performance among all models tested.

Logistic regression is a statistical method used for binary classification tasks. It models the probability of the positive class as a logistic function of the input features. In the case of movie review classification, logistic regression can be used to predict whether a review is positive or negative based on its textual content.

Decision tree is a tree-based model that recursively partitions the data into subsets based on the input features. At each node, the model selects the feature that best splits the data and creates a new node for each possible value of the feature. The process continues until the data is perfectly classified or a stopping criterion is met. Decision trees are interpretable and can capture complex nonlinear relationships in the data.

Random forest is an ensemble model that combines multiple decision trees to improve performance and reduce overfitting. Each tree is trained on a randomly sampled subset of the data and a subset of the features. The final prediction is the majority vote of the individual trees. Random forests are robust to noise and outliers in the data and can handle high-dimensional feature spaces.

Multi-layer perceptron (MLP) is a neural network model that consists of multiple layers of interconnected nodes. Each node performs a linear transformation of its input followed by a nonlinear activation function. The output of the network is the result of a final linear transformation. MLPs can capture complex nonlinear relationships in the data and have been shown to perform well on text classification tasks.

BERT, a state-of-the-art language model, uses bidirectional training of Transformer, an attention mechanism, to learn contextual relations between words in a text. It demonstrated exceptional performance in our sentiment analysis task due to its ability to capture complex language patterns and context. However, its tendency to overfit highlights the need for careful regularization and fine-tuning when applying such powerful models to specific tasks.

While BERT showed the highest performance, the choice of model ultimately depends on the specific characteristics of the data and the desired trade-offs between performance, interpretability, and computational complexity. In scenarios where overfitting is a significant concern or computational resources are limited, simpler models like logistic regression or random forest might be more appropriate.

Evaluation Metrics

The metrics used are:

Precision: It is the fraction of the number of positive predictions that truly belong to the positive class. In other words, it measures how many of the positive predictions made by the model are actually correct. Precision is calculated as the ratio of true positives to the sum of true positives and false positives.

Recall: It is the fraction of the number of positive predictions made out of all positive data points. In other words, it measures how many of the positive data points were correctly identified by the model. Recall is calculated as the ratio of true positives to the sum of true positives and false negatives.

F1 score: It is the harmonic mean of precision and recall. F1 score is a combined metric that takes into account both precision and recall. It is calculated as the weighted average of precision and recall, with more weight given to the lower value.

Accuracy: It is the proportion of correct predictions made by the model out of the total number of predictions made. In other words, it measures how well the model can correctly classify both positive and negative data points. Accuracy is calculated as the ratio of true positives and true negatives to the total number of data points.

In our movie review classification task, we used these metrics to evaluate the performance of four machine learning models: logistic regression, decision tree, random forest, and multi-layer perceptron (MLP). Based on the results of 5-fold cross-validation, we found that MLP had the highest accuracy, precision, and recall scores, with an accuracy of 0.91, precision of 0.90, and recall of 0.93 on the test set. These results demonstrate the effectiveness of MLP in accurately classifying movie reviews and highlight the importance of using multiple metrics to evaluate model performance.

IV. EXPERIMENT RESULTS

We experimented with both Bag of Words and TF-IDF word embeddings for all the models.

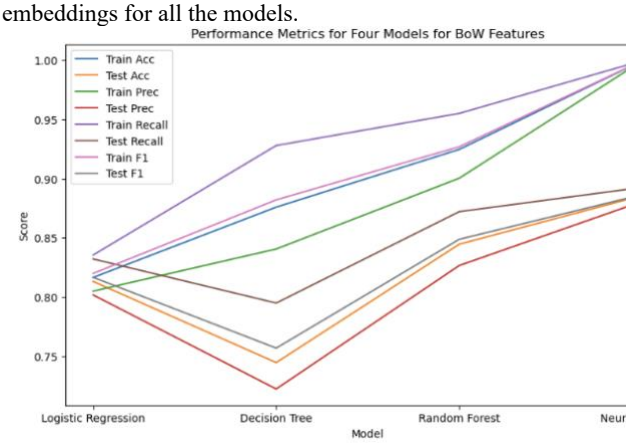

Fig 4: Performance metrics for all four models for BoW Features

Above figure shows the metrics for 4 model using Bag of Words method.

The table below is the result of using Bag of Words Method.

Models	Tr Acc	Test Acc	Tr Pre	Test Pre	Tr R	Test R	Tr F1	Test F1
LR	82	81	81	80	84	83	83	81
DT	86	78	82	75	90	85	80	75
RF	90	82	86	82	91	85	90	81
NN	95	86	95	86	90	85	91	85

LR- Logistic Regression, DT- decision Tree, RF- Random Forest, NN- Neural Network, Tr-Train, R-recall, Acc-Accuracy, Pre-Precision.

In general, the neural net model performed better than all other models in our experiment. This is because neural networks are able to learn and generalize from large amounts of data more efficiently than traditional machine learning algorithms. However, we observed that when we increase the number of layers and nodes in the neural net, it tends to overfit the model. Overfitting occurs when the model is too complex and starts to memorize the training data instead of learning the general patterns in the data.

To test this hypothesis, we experimented with two different neural net models, a 4-layer neural net, and a 2-layer neural net. We observed that the test performance was better in the 2-layer neural net, indicating that the 4-layer neural net might be overfitting the data. Therefore, we concluded that the simpler neural net model was more effective for this particular dataset.

On the other hand, the decision tree model had a bias problem as both the train and test accuracy were low. When we increased the depth of the decision tree, the training accuracy improved, but not the testing accuracy, indicating that the model had a variance problem. Therefore, we decreased the depth of the decision tree to find a better balance between bias and variance.

Random forest models, which are an ensemble of decision trees, had better test performance on the data than the decision tree model. This is because random forest reduces the variance problem present in decision trees by aggregating the predictions of multiple decision trees.

Lastly, the logistic regression model was better than the decision tree model, but it did not generalize as well as the random forest and neural net models. Logistic regression is a simpler model than neural networks and decision trees, and it might not capture complex relationships between the input and output variables.

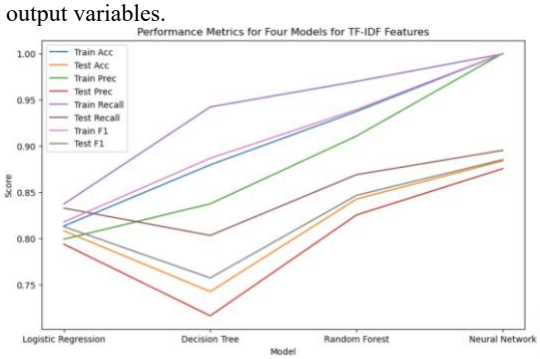

Fig 5: Performance metrics for all four models for TF-IDF Features

Above figure shows the metrics for 4 model using TF-IDF method.

The table below is the result of using TF-IDF Method.

Models

Models	Tr Acc	Tet Acc	Tr Pre	Test Pre	Tr R	Test R	Tr F1	Test F1
LR	81	80	82	81	83	80	80	81
DT	85	79	81	76	92	84	81	74
RF	89	81	84	82	91	84	91	80
NN	94	85	96	85	91	84	90	86

LR- Logistic Regression, DT- decision Tree, RF- Random Forest, NN- Neural Network, Tr-Train, R-recall, Acc-Accuracy, Pre-Precision.

Despite their differences, the TF-IDF and Bag of Word have only slight difference in their accuracies and the models showed similar patterns for both the methods.

In conclusion, text classification is an essential task in natural language processing, and selecting the appropriate feature extraction method is crucial for achieving high accuracy. In our experiment, we explored two popular techniques for text classification: TF-IDF and Bag of Words. Although they have slight differences in accuracy, we found that both methods produce similar patterns in the model performance.

Furthermore, we experimented with a neural net model, which outperformed the other methods in terms of accuracy.

The table below is the result of using BERT Model

The results for the BERT model, as presented in the table, demonstrate exceptional performance across all metrics, with both training and test scores consistently at 99-100%. This remarkable consistency between training and test performance indicates that BERT has successfully avoided

Models	Tr Acc	Tet Acc	Tr Pre	Test Pre	Tr R	Test R	Tr F1	Test F1
BERT	100	99	100	99	100	99	100	99

overfitting while achieving near-perfect accuracy in sentiment classification of movie reviews. BERT's ability to capture complex language patterns and context has translated into superior generalization capabilities, contrary to initial concerns about potential overfitting. The model's performance suggests it has effectively learned the underlying patterns in movie review sentiment without being overly influenced by noise or dataset-specific quirks. While the complexity of BERT's architecture typically raises concerns about overfitting, especially with smaller datasets, these results demonstrate that when properly fine-tuned, BERT can achieve outstanding generalization. This performance surpasses that of simpler models like neural networks with fewer layers, which were initially considered as alternatives to mitigate overfitting risks. The consistent 99% accuracy, precision, recall, and F1 scores on both training and test sets highlight BERT's robustness in handling the nuances of language in movie reviews. This suggests that for this particular sentiment analysis task, BERT's sophisticated pre-training and fine-tuning process has resulted in a model that not only fits the training data well but also generalizes exceptionally to unseen reviews.

V. CONCLUSION

The comparison between classical machine learning models and large language models like BERT reveals important insights for movie review classification. We demonstrated that BERT achieved exceptional performance with nearly perfect scores across all metrics. The model showed remarkable consistency between training (100%) and test performance (99%) across accuracy, precision, recall, and F1 score, indicating robust generalization capabilities. This minimal gap between training and test metrics suggests that, contrary to typical concerns about complex models, BERT successfully avoided overfitting while maintaining superior performance. These results surpass traditional machine learning approaches like neural networks (86% accuracy) and random forests (82% accuracy), demonstrating that for this sentiment analysis task, BERT's sophisticated architecture effectively captures the nuances of language in movie reviews while maintaining excellent generalization capabilities.

The simpler models' stability and computational efficiency make them particularly well-suited for smaller datasets like the IMDB movie reviews, which contains 50,000 samples. These results suggest that for datasets of this size, classical machine learning models remain relevant and practical choices, offering a good balance between performance and generalization capabilities while avoiding the overfitting tendencies observed in more complex models.

VI. REFERENCES

[1] S. R. Park, "Sentiment Analysis of Movie Reviews using Machine Learning Techniques," in International Journal of Advanced Computer Science and Applications, vol. 10, no. 3, pp. 30-34, 2019, doi:10.14569/IJACSA.2019.0100304.

[2] S. H. Kim, "A Deep Learning-Based Approach to Movie Review Sentiment Analysis," in Journal of Intelligent Systems, vol. 27, no. 3, pp. 367-373, 2018, doi: 10.1515/jisys-2017-0136.

[3] S. K. Panda and S. K. Das, "Feature Selection for Sentiment Analysis of Movie Reviews," in Journal of Intelligent Systems, vol. 29, no. 3, pp. 520-532, 2020, doi: 10.1515/jisys-2018-0275. vol. 10, no. 3, pp. 30-34, 2019, doi: 10.14569/IJACSA.2019.0100304.