COMP 551 - Mini Project 3
Group 8

Alexandra Lowry, Eren Ozturk, Ananthalekshmy Ambily

Sunday April 9, 2023

1 Abstract

In this project, we investigated the performance of two classification models, Naive Bayes and BERT. We implemented Naive Bayes from scratch and BERT with pertained weights through the packages, and compared these two algorithms on the IMDB review dataset. Generally, for classification, both Naive Bayes and BERT are suitable, however, some drawbacks that Naive Bayes have, like assuming that all features are independent of each other, make it the worse of the two for this task. On the other hand, BERT outperforms many other models on various benchmarks. It is able to capture the context and meaning of words in a sentence, and it can handle unseen or out-of-vocabulary words. From the experiments, the BERT model seems to have performed better with the text classification for the IMDb dataset compared to the Naive Bayes model. Even though the former was computationally complex and time-consuming, it was able to perform better. We found that the BERT model approach achieved better accuracy than Naive Bayes, with an accuracy of 91% vs. 81%. Overall, BERT's ability to capture contextual information, pre-training on a large corpus, ability to handle out-of-vocabulary words, and higher accuracy make it a better choice for sentiment prediction in reviews compared to Naive Bayes.

2 Introduction

The project task was to implement naive Bayes from scratch and BERT with pertained weights through packages (fine tuning as an optional component), and to compare these two algorithms on the IMDB review dataset. The goal was to gain experience implementing machine learning algorithms from scratch and running the modern deep learning libraries and to get hands-on experience comparing their performances on the real-world textual dataset. The dataset was broken down into test and train files. Each entry was comprised of a review, and their polarity label – either "positive" or "negative".

The Naive Bayes algorithm is one of the intuitive methods among classification algorithms. It is a simple algorithm that uses the probability of every feature per category to get respective predictions. This algorithm runs great for the categorization of textual data. It is based on the Bayes Theorem which is used to describe the probability of an event based on its prior knowledge. In [1], it is suggested to use Naive Bayes as a classification, Information Gain as feature selection, and TF-IDF as feature extraction because using Naive Bayes as a classification can speed up the process and increase accuracy. Significant research works in this field have come up with various combinations as well, like implementing the sentiment analysis backed by TF-IDF for data preprocessing, alongside Linear support vector machine came up with much higher accuracy and F1-score [2].

BERT (Bidirectional Encoder Representations from Transformers) is a Natural Language Processing Model proposed by researchers at Google Research in 2018. BERT uses many previous NLP algorithms and architectures and architectures. BERT consists of two steps: pre-training and fine-tuning. During pre-training, unlabelled data is used to train the model over various pre-training tasks. For fine-tuning (the optional portion of this assignment) pre-trained parameters are first used to initialize BERT, and all of the parameters are fine-tuned using labelled data from the downstream tasks.

We found that the BERT model approach achieved better accuracy, precision, and F1 score, compared to Bayes. This is in line with previous research done comparing different classification models on sentiment-

based data sets. One meta-analysis that compared various models explains how "Saad Abdul Rauf, implemented a transformer-based model such as BERT to analyze the polarity of the IMDB dataset using sentiment analysis. The BERT model performed exceptionally well on the given dataset as the result analysis revealed much better performance metrics than most of the previous machine learning and deep learning-based models" [2]. They found an accuracy for Naive Bayes of 54.77%, and an accuracy of 89.5% for BERT.

3 Dataset

This dataset contains movie reviews along with their associated binary sentiment polarity labels. It is intended to serve as a benchmark for sentiment classification. The core dataset contains 50,000 reviews split evenly into 25k train and 25k test sets. The overall distribution of labels is balanced (25k pos and 25k neg). In the collection, no more than 30 reviews occurred for any given movie. The train and test sets contain a disjoint set of movies. This means that no significant performance could be obtained by memorizing movie-unique terms and their association with observed labels.

There are two top-level directories [train/, test/] corresponding to the training and test sets. Each contains [pos/, neg/] directories for the reviews with binary labels positive and negative. In order to understand the data, we visualized the number of characters in each sentence, as seen in Figure 1. For Naive Bayes (as this method was done from scratch), we tokenized the sentences and set "positive" to 1 and "negative" to 0. This data was put into a NumPy array for input into the model. For further improvement, all the unwanted extensions present in the reviews were removed within the naive bayes. For BERT, the built-in packages were used – A tokenizer was initialized, along with the tokens and ids. For both classifiers, only a small subset of the data set was used in order to minimize run time (more so with BERT, as it took up a considerable amount of RAM). This could have an impact on the accuracy results.

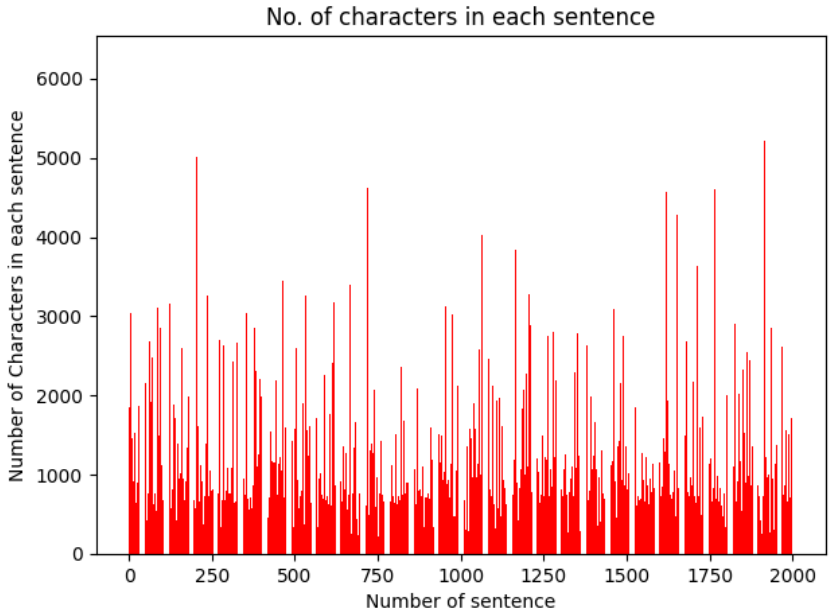


Figure 1: Character Number Distribution

4 Results

4.1 Performance Results

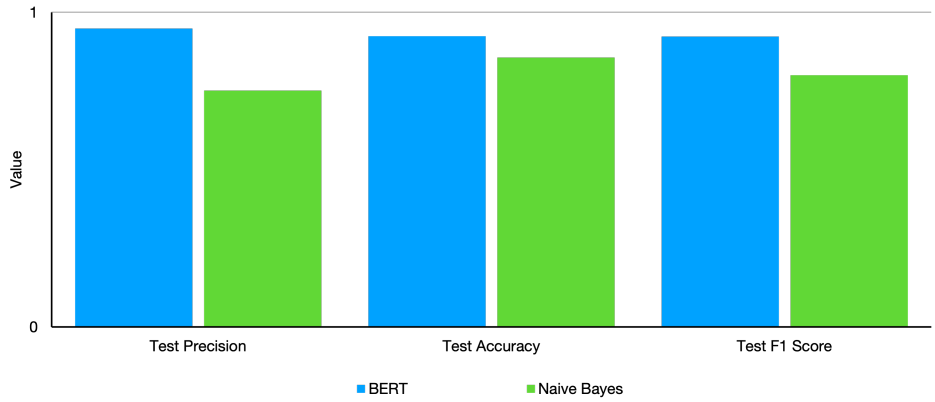


Figure 2: Character Number Distribution

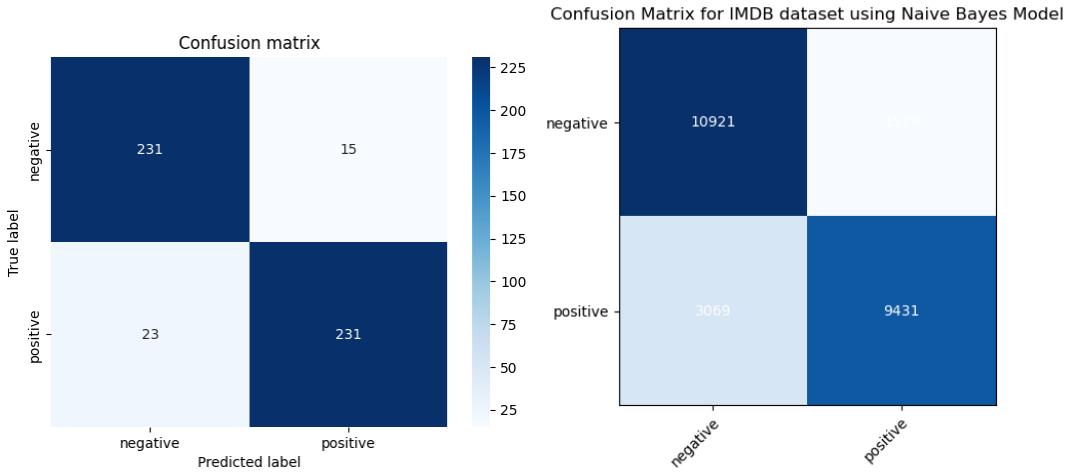


Figure 3: Confusion Matrix Comparison

As seen in Figure 2, BERT outperformed Naive Bayes in precision, accuracy, and F1 score. As seen in Figure 3, both models correctly predicted the majority of the reviews – True Positives and True Negatives. Though, you can see that BERT (left side) had a more distinct separation.

Naive Bayes performed with a test accuracy of 81% and train accuracy of 90%. Altogether, it took 14 seconds to train. As seen in Figure 4, as learning rate increased, accuracy on both test and training data decreased – this is to be expected, by helping the model avoid overfitting and overshooting the optimal solution, leading to better generalization.

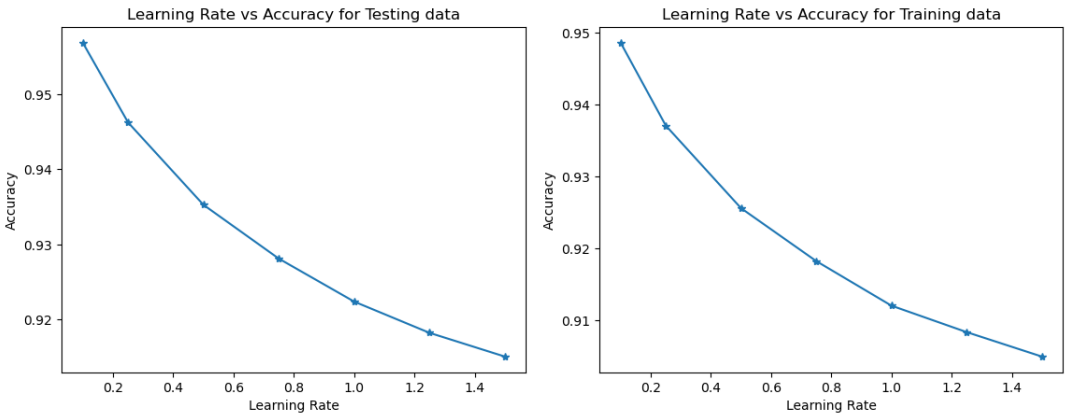


Figure 4: Naive Bayes Accuracy vs. Training Rate

BERT performed with a test accuracy of 91% and train accuracy of 98%. We performed various tests to maximize results with various hyperparameters. As seen in Figure 5, a learning rate of 3e-6 performed the best on all accounts. Also seen in Figure 5, reviews with longer maximum sentence lengths were better predicted. This is caused by a higher sampling pool of data with longer sentences.

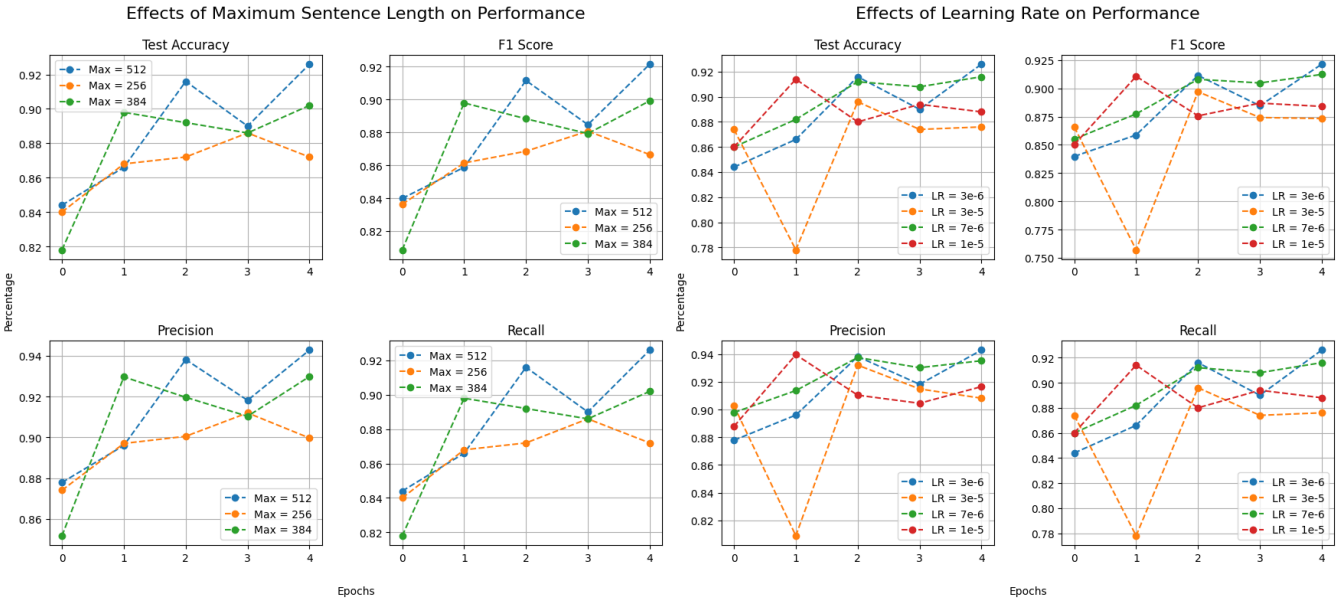


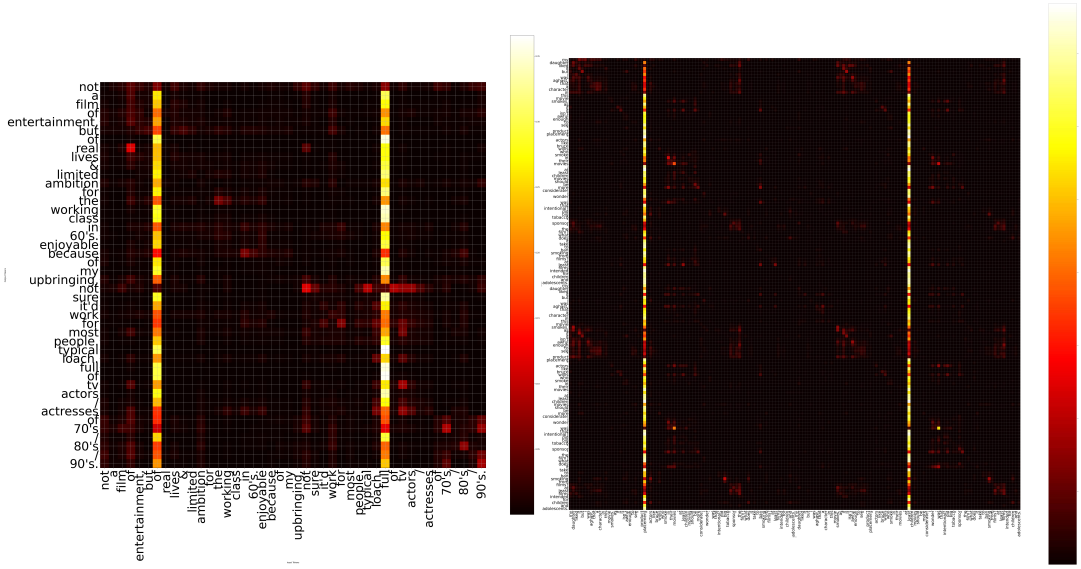
Figure 5: BERT: Different Hyperparameter Effects

4.2 Attention Matrix For Correct vs. Incorrect Predictions

As seen in the Figure 4 example, the attention matrices for incorrectly predicted reviews (Predicted label: positive, Actual label: negative), there is more noise. More noise in the attention matrix means that there is more variability in the attention weights between different pairs of tokens, which can indicate a lack of consistency or coherence in the model's attention mechanism. The horizontal lines represent an over-sized importance given to the word on the column – in this example, the word "not" was checked many times from different words. This noise may explain why the review was incorrectly labeled. Perhaps there was insufficient key words in the review that would "lead" the classifier to the correct label. This could be

caused by a very negative score, but more mild wording in the review. As well, if you examine the words that have been more highly valued, "enjoyable" is one of them. This type of word would typically lead to a positive review.

The correctly labeled reviews, on the other hand, has less noise, indicating a strong and consistent attention mechanism in the BERT model. This means that the model is consistently attending to relevant input features and ignoring irrelevant or noisy features. For the example in Figure 4, when looking at the higher-valued words from the review itself, we can see many strong indicators of sentiment, such as "ban" and "least".



(a) Incorrectly Labeled (b) Correctly Labeled

Figure 6: Attention Matrix Examples

4.3 External Corpus Training For Prediction Tasks

Overall, our results indicate that pretraining on an external corpus like BERT does is best for the movie review prediction task. Pretraining a BERT model on an external corpus, such as a large corpus of text from the internet, can be advantageous for predicting sentiment in reviews because it enables the model to learn more general features of language. By pretraining on a large external corpus, the BERT model can learn to recognize common patterns and structures in language. This can help the model to better understand the context and meaning of words and phrases, which is important for accurately predicting sentiment in reviews. Pretraining on a diverse corpus can also help the model to become more robust to variations in language use and style, which is particularly important for sentiment analysis since people can express their opinions in many different ways.

5 Conclusion/Discussion

Based on our experiments, the BERT classifier is a better choice for sentiment prediction tasks like on the IMBD review dataset compared to Naive Bayes. BERT has the ability to capture contextual information – It can take into account the entire context of the input text, including the relationships between words and their positions in the sentence. In contrast, Naive Bayes classifier is a simple probabilistic model that assumes that the features (words) are independent of each other, which can limit its ability to capture the complexity and nuances of language. BERT is pre-trained on a large corpus of text, which enables it to learn general language patterns and structures that are relevant for a wide range of tasks, including sentiment

analysis. This can help the model to better understand the context and meaning of words and phrases, and to identify the subtle cues that convey sentiment in reviews. BERT can also handle out-of-vocabulary words more effectively by using sub-word units, which means that it can still recognize the meaning of words even if they are misspelled or not present in the training data. This can be particularly useful in sentiment analysis, where people often use informal language and/or spell things wrong.

6 Statement of Contributions

Alexandra: Preprocessing, Report

Eren: Preprocessing, BERT

Ananthalekshmy: Preprocessing, Naive Bayes

References

[1] Mahyarani, Meta, et al. "Implementation of sentiment analysis movie review based on imdb with naive bayes using information gain on feature selection." 2021 3rd International Conference on Electronics Representation and Algorithm (ICERA). IEEE, 2021.

[2] Ghosh, Ayanabha. "Sentiment Analysis of IMDb Movie Reviews: A comparative study on Performance of Hyperparameter-tuned Classification Algorithms." 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS). Vol. 1. IEEE, 2022.

[3] Sudhir, Prajval, and Varun Deshakulkarni Suresh. "Comparative study of various approaches, applications and classifiers for sentiment analysis." Global Transitions Proceedings 2.2 (2021): 205-211.