CP322: Mini project 2

Alexandros Ioannou, Duc Nguyen Minh, Florian Novak, Saumya Patel

I. Abstract

In this project, we addressed multi-class classification problems regarding text data. For our work, we focused on two datasets: the 20 Newsgroup dataset for general news classification and the IMDB dataset for sentiment analysis of movie reviews. Our approach involved analyzing the two datasets and implementing and comparing five different models: Logistic Regression, Decision Tree Classifier, Linear Support Vector Classifier (SVC), AdaBoost Classifier, and Random Forest Classifier. Our analysis involved performing feature selection on the 20 Newsgroup dataset and plotting out basic distributions to further improve our understanding on the dataset; and cleaning/tidying of the IMDB dataset. The highest accuracy on the 20 Newsgroup dataset was achieved by the Linear SVC with 69.28%, while the highest accuracy on the IMDB dataset was achieved by Logistic Regression with 87.89%.

II. Introduction

Our project can be divided into four different stages of work for every dataset plus the implementation of the models in general. First, we loaded both datasets into Kaggle with the help of the *sklearn* library. In the next step, both datasets had to be checked and cleaned. For the 20 Newsgroup dataset, feature selection was performed, in which we concluded with following categories: 'alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc', 'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware', 'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles', 'rec.sport.baseball', 'rec.sport.hockey', 'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space', 'soc.religion.christian', 'talk.politics.guns', 'talk.politics.mideast', 'talk.politics.misc', 'talk.religion.misc''. For the IMDB dataset, the train and test files were selected, in which string manipulation was performed to obtain a clean and optimal output. Next, for both of the datasets, we split the data into training and testing data accordingly to implement the models. After all the preparation, with the help of sklearn, we implemented the five different models: Logistic Regression, Decision Tree Classifier, Linear SVC, AdaBoost Classifier, and Random Forest Classifier and created classification reports in order to better analyze our results. Lastly, we select the best classifier for every dataset.

III. Related Work

Existing research has already examined work on different multi-class classification problems with text data and like this, has given us some inspiration for working on these kinds of problems. For example, Rabbimov, I. M., and S. S. Kobilov (2020)[1] worked on text classification of Uzbek news articles, similar to our news dataset, and compared the performance of six different classifiers. They mention the importance of preprocessing text data to be able to better work with the data, like filtering out stop words. Due to this, we also looked a lot into pre-processing the text data before even working on the final models. Karimi et al. (2021)[2] worked on a similar problem, trying to classify text documents specifically out of the human medicine domain. They explicitly state the importance of choosing a balanced training data set

with correctly labeled data to be able to train the models in a neutral way. This problem did not appear in our work, as we used the pre-defined well-known datasets. Both of the above-mentioned works managed to get a classification model with a prediction accuracy of above 80% for their datasets because of trying all kinds of different classification models. That was also our inspiration to use many different classification algorithms and compare them to each other, trying to find the best prediction model for each dataset.

IV. Datasets

The 20 Newsgroup dataset consists of approximately 20,000 newsgroup documents, categorized into 20 different groups. The IMDB dataset includes movie reviews in a text format, labeled based on sentiment (positive or negative). For both datasets, we implemented standard preprocessing methods, including text normalization, tokenization, removal of stop words and vectorization using TF-IDF. These preprocessing steps were done to transform raw text into a structured form suitable for the development of the models.

V. Proposed Approach

In terms of feature engineering, we employed TF-IDF vectorization, which transforms text into a meaningful representation of numbers. This technique highlights the importance of specific words relative to the dataset.

After pre-processing both datasets, we were ready to start to develop five different classification models. Our primary models included Logistic Regression, Decision Tree Classifier, Linear SVC, AdaBoost Classifier, and Random Forest Classifier. As both datasets already had a training dataset and a testing dataset with it, there was no need to further split the data for training purposes. Next, we started to implement the models:

Logistic Regression: Logistic Regression was chosen for its efficiency and effectiveness in binary and multiclass classification. It's particularly well-suited for datasets like IMDB reviews, where the goal is to categorize texts into predefined classes (positive or negative).

Decision Tree Classifier: This model was included for its interpretability and ease of use. Decision trees work by creating a model that predicts the value of a target variable by learning simple decision rules inferred from data features.

Linear SVC: Linear SVC was selected for its ability to handle large feature spaces, as is common in text classification tasks.

AdaBoost Classifier: This ensemble method combines multiple weak classifiers to create a strong classifier. AdaBoost was used to see if boosting could enhance the performance of basic classifiers.

Random Forest Classifier: As an ensemble of decision trees, this model is known for its high accuracy and ability to run in parallel.

Overall, the selection and implementation of these algorithms were used to compare methods in text classification, assessing their strengths and weaknesses in handling complex datasets.

VI. Results

20 Newsgroup Dataset

Following our model implementation and assessment, Linear SVC had the highest accuracy on the 20 Newsgroup dataset with 69.28% which marginally outperformed Logistic Regression with an accuracy of 69.09%. The other models followed, the Random Forest Classifier achieved an accuracy of 62.51%, the Decision Tree Classifier achieved 44.13% accuracy, and lastly, the AdaBoost Classifier achieved 36.59% accuracy. In summary, Linear SVC had the highest accuracy and AdaBoost had the lowest accuracy in this dataset. Figure 1 below shows the comparison in performance of each model accordingly. The runtime of this dataset was reasonable since feature selection was performed and the data provided was considerably less than the IMDB dataset.

In the context of the 20 Newsgroup dataset with many categories, the role of TF-IDF is to provide the nuanced feature set that helps the models to distinguish between the various topics. When we reduced the problem to only four categories, the models could achieve higher accuracy because the TF-IDF scores would have created a feature set with better separation between categories. As the number of categories increases, the overlap in significant terms across different topics can make classification more challenging, leading to the decrease in accuracy. TF-IDF is a tool to mitigate this, but its effectiveness can be limited by the complexity of the task.

IMDB Dataset

On the IMDB dataset, Logistic Regression had the highest accuracy at 87.89%, with Linear SVC close behind at 87.49%. The other models followed, the Random Forest Classifier achieved an accuracy of 83.92%, the AdaBoost Classifier achieved 80.43% accuracy, and lastly, the Decision Tree Classifier achieved 70.61% accuracy. In summary, Logistic Regression had the highest accuracy and Decision Tree had the lowest accuracy in this dataset. Figure 2 below shows the comparison in performance of each model accordingly. The runtime on this dataset was significantly longer compared to the 20 Newsgroup dataset due to the large input files that contained 50,000 movie reviews in total training and testing.

K-Fold Cross-Validation

Ran the cross validation for both datasets with a k-value of 5. For the 20 Newsgroup Dataset logistic regression had an average cross validation accuracy of 73.92%. Decision tree had an average accuracy of 47.66%. Linear SVC had an average accuracy of 76.01%. AdaBoost had an average accuracy of 39.59%. Random forest had an average accuracy of 67.02%. For the IMDB dataset logistic regression had an average accuracy of 87.65%. Decision Tree had an average accuracy of 70.11%. Linear SVC had an average accuracy of 86.81%. AdaBoost had an average accuracy of 80.25%. Random forest had an average accuracy of 83.75%. Figures 3 and 4 below show a comparison of the average cross-validations across all models.

VII. Discussion and Conclusion

This project illuminated the efficacy of various machine learning models in text classification, particularly highlighting the strengths of Logistic Regression and Linear SVC across diverse datasets. The preprocessing steps, like tokenization and TF-IDF vectorization, were crucial in transforming raw text into analyzable data, significantly impacting model performance. Notably, the project revealed that while some models excel in certain contexts (e.g., Logistic Regression in both datasets), others might require more fine-tuning (like Decision Tree Classifier). On average, the IMDB dataset outperformed the 20 Newsgroup dataset across all models.

VIII. Statement of Contributions

- **Alexandros Ioannou:** Implemented models and plots on the IMDB dataset and came up with results and conclusions. Contributed to the write-up of all sections.
- **Duc Nguyen Minh:** Implemented models and plots on the 20_newsgroup dataset. Contributed to the project write-up
- **Florian Novak:** Responsible for research on related work. Contributed to data preparation. Contributed to write-up.
- **Saumya Patel:** Implemented k-fold validation for both datasets and for all the models, cleaned and prepared the IMDB dataset, and helped with the write-up.

References

1. Rabbimov, I. M., and S. S. Kobilov. "Multi-class text classification of uzbek news articles using machine learning." *Journal of Physics: Conference Series*. Vol. 1546. No. 1. IOP Publishing, 2020.
2. Karimi, Kamran, et al. "Classifying domain-specific text documents containing ambiguous keywords." *Database* 2021 (2021): baab062.

Appendix

Figure 1

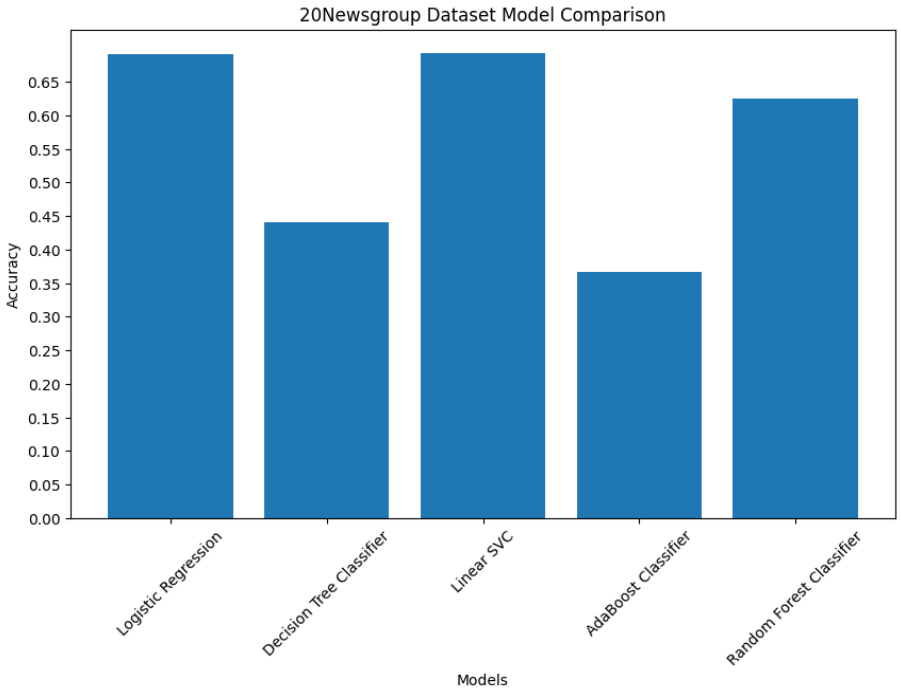

Figure 2

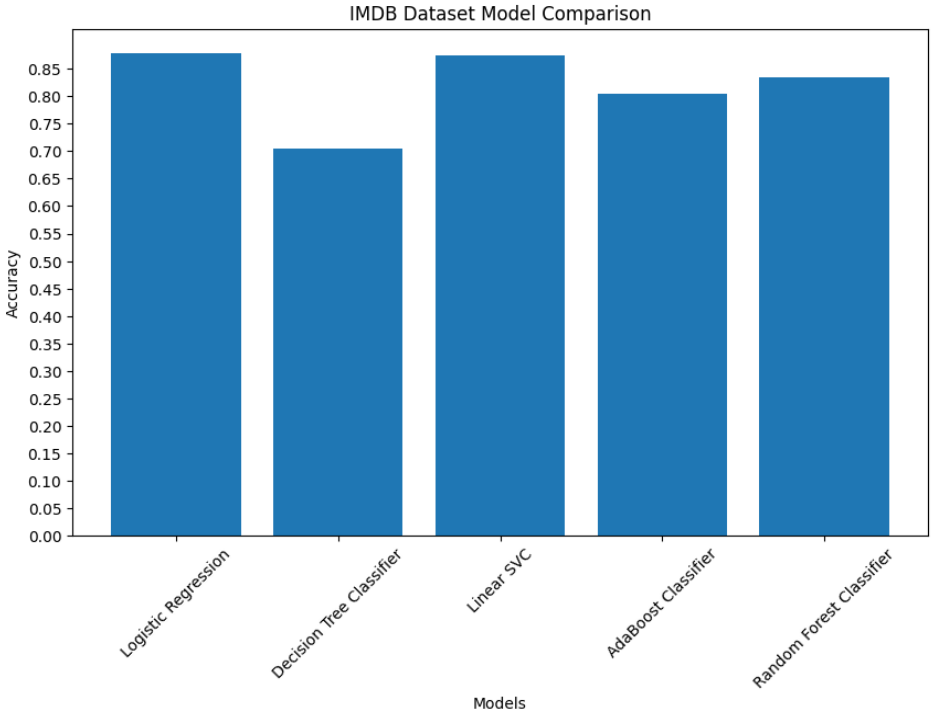

Figure 3

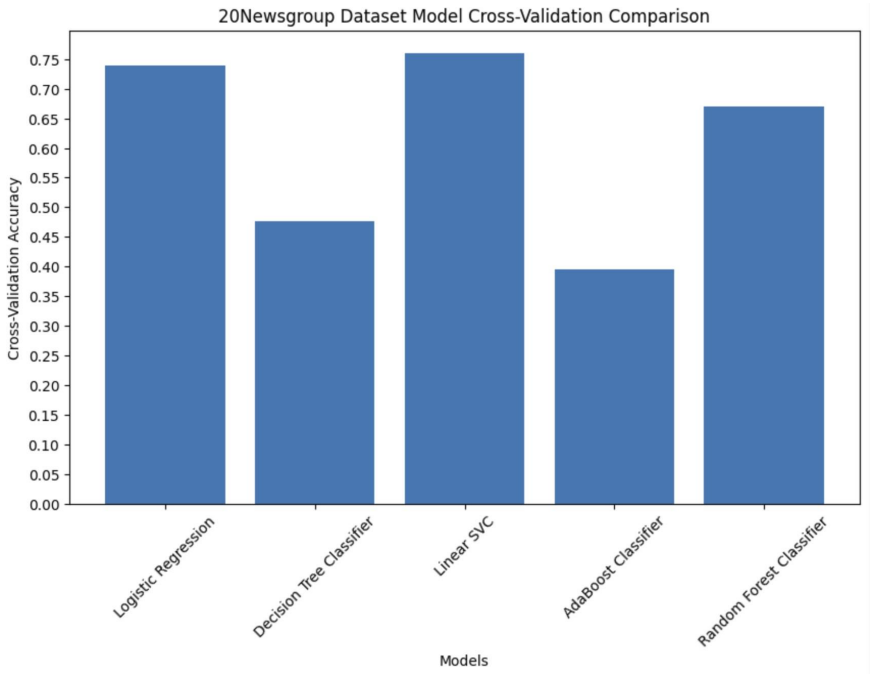

Figure 4

IMDB Dataset