Online Sexism Detection Report

Khondokar Mohammad Ahanaf Hannan-20101079

Humayra Musarrat-20101089

Nishat Zerin-20101136

Sameer Sadman Chowdhury-21101105

BRAC University, CSE

Submitted To

Dr. Farig Yousuf Sadeque

Assistant Professor, BRAC University

11/5/23

1 Introduction

In our project, we focused on classifying sexist and non-sexist language using the help of three different models, which are Logistic Regression, support vector machine (SVM) and neural network. While SVM is a non-linear model that has been demonstrated to perform well on text classification tasks, logistic regression is a linear model that is frequently employed in NLP applications. Neural networks have displayed astounding performance on a variety of NLP tasks, especially in deep learning models. Our report also goes into detail about our reasoning behind why we chose these models as the most effective way of completing our objective. To decide which model performed the best, we looked at its macro average and weighted score. In order to detect and lessen the use of sexist language in various forms of communication, our study intends to develop models that can accurately categorize language as sexist or non-sexist.

2 Data Exploration

We have explored our dataset using a variety of commands. At first we checked the first few rows of the dataset using the head() method to make sure we implemented the correct data set. Consequently, we checked the shape of the data structure, data types of each column and the summary statistics of numerical columns to grasp a good understanding about the dataset as we may need this information later in the code. While data exploration, we found extremely important information about our dataset. Our dataset consists of two categories, sexist and non-sexist. We found out that the non-sexist category largely dominated our dataset. This made us come to an interesting conclusion, which is the fact that our non-sexist data would be more efficiently trained and our model class can predict non-sexist data with a much higher accuracy than sexist data. Finally, we checked for null values in our dataset to avoid complications while model training. No null values were found hence further data exploration was not required.

3 Text Pre-processing

We have achieved text preprocessing for all the models with a series of steps. At first, we removed all punctuation and converted all the text to lowercase. Then, we removed all the digits from the text using regex. Next, we tokenized the text and removed all the stop words from the text. Subsequently, we performed lemmatization on the words in the text. Lastly, the words were joined back together into a string with spaces between them and the resulting preprocessed text is returned. Overall, this function takes in text data and returns a preprocessed version of the text data that can be used for further text analysis tasks.

4 Feature Extraction

TF-IDF vectorization is performed to convert the preprocessed text into numerical features. It converts our textual data into a format that can be easily used for machine learning. By using TF-IDF vectorization, we can capture the unique characteristics of each document, which helps to improve the performance of machine learning models trained on textual data. Additionally, this technique can also help to reduce the impact of commonly occurring words that are not informative

5 Methodology

Logistic Regregssion

A logistic regression model is trained on the training data and used to predict labels for the testing data. The code calculates evaluation metrics such as accuracy, confusion matrix, and classification report. Additionally, visualizations of the classification report and confusion matrix are generated using heatmaps. The code also includes loading development data, preprocessing it using the same steps as the training data, making predictions, converting predictions to labels, creating a DataFrame with the predicted labels, and saving the results to a CSV file.

In summary, this code demonstrates the process of text classification using logistic regression by performing data preprocessing, feature extraction, model training, prediction, evaluation, and visualization.

A useful methodology for distinguishing between sexist and nonsexist data is logistic regression. This is accomplished by training a decision boundary that divides the two classes according to input features. In order to classify texts, a model must first examine preprocessed text data, extract numerical features (using, for example, TF-IDF), and train on data that has been labeled in order to learn the link between features and labels. It can then forecast whether future occurrences will be sexist or not. The model turns probabilities into binary predictions by choosing a threshold. Metrics for evaluation measure how well the model can distinguish between the two classes. In general, logistic regression uses the correlation between labels and input attributes to categorize data as sexist or nonsexist.

Support Vector Machine

Support Vector Machine (SVM) is a popular machine learning algorithm that classifies text data into different categories based on the features extracted from the text. It works by finding the hyperplane that best separates the data points belonging to different categories. During training, the SVM algorithm finds the hyperplane that maximizes the margin between the data points of different categories After training, the SVM model computes a score for each category, and the category with the highest score is considered to be the predicted category for the given text.

It can be a suitable model for differentiating between sexist and non-sexist

data. Firstly, SVMs work well with high-dimensional feature spaces like our data. Our input data is high-dimensional, with each feature representing a word or a combination of words. Moreover, SVMs are known to work well in such feature spaces, allowing for accurate classification. Secondly, SVMs are robust to noise which means it can ignore noisy data such as misspellings, slang, and other non-standard language forms. Thirdly, SVMs can handle non-linear relationships as well. In our dataset, the relationship between the features and the target variable is non-linear. As SVMs can handle these issues well it is an effective choice for classifying our dataset.

Neural Network

In machine learning, neural network models are frequently employed because of their ability to recognize intricate non-linear correlations, manage highly dimensional input, and learn hierarchical representations. They excel at jobs involving unstructured data, have automatic feature extraction, and gain from transfer learning and pre-trained models. Large-scale datasets may be processed using neural networks, which can also adapt and learn continuously. On the other hand, they could be challenging to read and may be computationally expensive and require a lot of training data.

Neural Network models are useful for differentiating between sexist and non-sexist data because they can capture complex patterns, learn hierarchical representations, and handle high-dimensional text data. They excel at capturing non-linear relationships in language and can identify discriminatory patterns. By learning representations of text data, they extract relevant features and understand the underlying meaning. Neural networks are flexible, adaptable to different language styles and sources, and can generalize well to new examples. Their large-scale data handling capabilities, continual learning, and the use of pre-trained models further enhance their effectiveness in detecting sexism in language.

6 Result and Analysis

After testing each model on the test set, we obtained the classification reports containing precision, recall, f1-score, accuracy, macro average and weighted average for each model.

Our model has 10602 non-sexist data and 3398 sexist data. We can see a a huge imbalance between these two categories. This imbalance inadvertently means that our model would be trained more with non-sexist data. Hence our models would be much better at predicting non-sexist data than sexist data. As a result, all our results would be biased towards non-sexist data. In order to prevent this, we only took macro average and weighted average into consideration. Precision, F1 and recall were not prioritized in our project.

The logistic regression model achieved an f1 accuracy score of 82% and performed well in predicting non-sexist data with a precision of 81% and an F1-score of 89%. However, it did not perform well in predicting sexist data with a precision of 85% and an F1-score of only 45%. The macro average has a precision of 83% with a f1-score of 67% but the weighted average is better with a precision of 88% and f1-score of 78%.

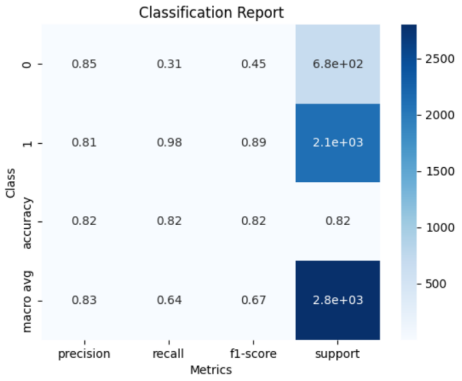

Figure 1: Classification Report

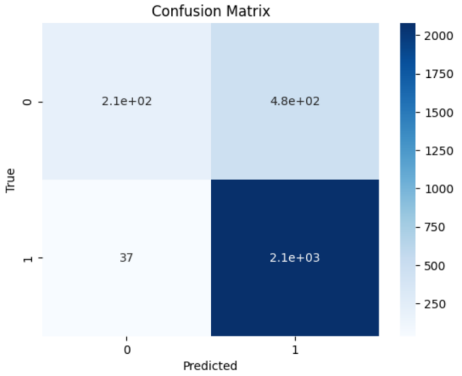

Figure 2: Confusion Matrix

The **SVM model** also achieved an overall accuracy of 82%, but it performed better in predicting both sexist and non-sexist data with a precision of 81% and 95% respectively. Its f1-score for non-sexist data was 90%, while for sexist data, it was 45%. The macro average of this model has a precision of 88% with a f1-score or 67% while the weighted average has a precision of 85% with a f1-score of 79%

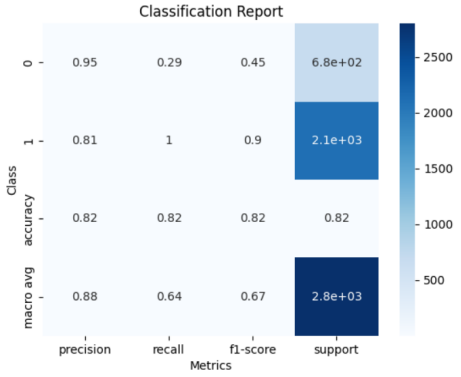

Figure 3: Classification Report

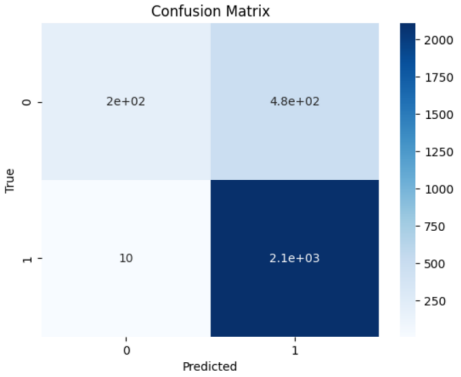

Figure 4: Confusion Matrix

The **neural network model** achieved an accuracy of 77%, which is slightly lower than the other two models. Its precision for non-sexist data was 83%, while for sexist data, it was only 56%. The f1-score for non-sexist data was 85%, while for sexist data, it was 51%. The macro average of this model has a precision of just 70% with a f1-score of 68% and the weighted average has a precision of 76% with a f1-score of 77%.

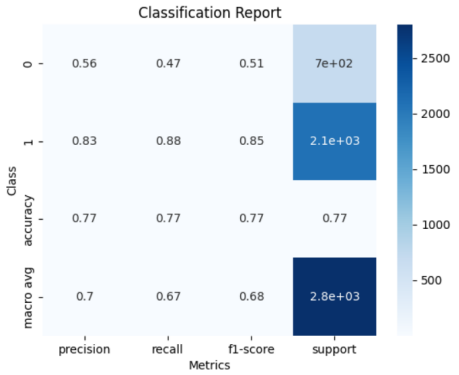

Figure 5: Classification Report

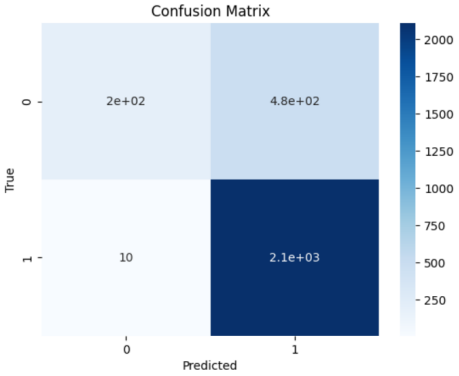

Figure 6: Confusion Matrix

The SVM model performed the best, with a weighted average of 0.79. This suggests that the SVM model was better at correctly identifying the sexist data, which was the primary objective of our project. Therefore, we can conclude that the SVM model would be the most suitable model for this task.

Based on the classification reports for the logistic regression, SVM, and neural network models, we can observe that the logistic regression and SVM models have similar performance, with a macro average of 0.67. The neural network model has a slightly higher macro average of 0.68.

In terms of weighted average, the logistic regression model has the lowest score of 0.78, while the SVM and neural network models have higher scores of 0.79 and 0.77 respectively.

Overall, the SVM model appears to have the best performance. However, it is important to consider other factors such as model complexity, training time, and computational resources required when choosing the best model for a particular use case.

7 Conclusion

In conclusion, our project aimed to classify text data into either sexist or non-sexist categories using three different machine learning models: logistic regression, support vector machine (SVM), and neural network. After preprocessing the data and extracting features, we trained and tested each model on our dataset.

The SVM model had the greatest F1 score of 0.79, indicating that it was best at balancing precision and recall. Our results revealed that all three models were able to attain excellent accuracy in identifying sexist and non-sexist data.

In general, our experiment showed how well machine learning models work for classifying text input into delicate categories like sexism. In the long run, we wish to contribute to a more equal society by using our work to increase the precision of future models that are comparable to ours.