Hate Speech: Cyberbullying Classification

Abbiegael Chu

AJ23 SYD009

TABLE OF CONTENTS

INTRODUCTION

Hate speech has run rampant in recent times and has been further spread through the availability of computers and smartphones. The anonymity that our computers provide gives some people the courage to verbally attack people through online channels, which is known as cyberbullying. Cyberbullying is not only sharing hateful words through online channels, but it also diminishes the psyche of the person receiving the hate.

Fortunately, natural language processing (NLP) is a branch under artificial intelligence that processes languages and text. NLP can help organizations classify texts into predefined classes, perform sentiment analysis, keyword spotting, and do contextual analysis on the body of text. In the context of this report, NLP is used to help classify cyberbullying hate text collected from Twitter into different types. Machine learning techniques were implemented and evaluated in this report, and the best performing model was recommended.

LITERATURE REVIEW

LOGISTIC REGRESSION

According to (Géron, 2017), logistic regression is utilized when it is needed to estimate the probability of an instance, which is a class. An example given was if an email was considered spam or not. If the estimated probability exceeds the threshold set, then the model will predict that the instance belongs to the class. The regression is an S-shaped sigmoid function that gives an output from 0 to 1.

Logistic regression is a popular and basic algorithm for classification. Recently, (Avanthika, Mrithula, & Thenmozhi, 2023) proposed a multimodal approach to identify hate speech and its targets. The main algorithms used in this case were logistic regression and support vector machines (SVM), which were used to process text data from social media.

Other than multimodal machine learning approaches, (Anjum & Katarya, 2023) used deep learning to detect hate speech. Bidirectional Encoder Representations from Transformers (BERT) was mainly used to determine the nature of the tweet. The sentiment measurement for the study used logistic regression with a ReLu activation function to check whether the text was hate or non-hate.

RANDOM FOREST CLASSIFIER

From (Géron, 2017), random forest is an ensemble method based on decision tress. Random forest, as the name suggests, is a cluster of decision trees with small alternations from each other to prevent overfitting. Random forest can be a regressor that churns out continuous outputs, or a classifier that gives discrete outputs which are applied to classification tasks.

(Ndenga, 2023)'s study explores a deep decision forests model that detects hate speech in textual twitter data. The author created three models. The first two models were gradient boosted trees and random forest, while the third model added a universal sentence embedding step in the gradient boosted trees method. Overall, the gradient boosted trees with the universal sentence embedding step performed the best, with the highest accuracy, recall, precision and recall among the three models.

According to (Das, Bhattacharyya, & Sarkar, 2023)'s study, SVM, Decision Tree and Random Forest had the best performance compared to other classification models, like Logistic Regression, Gaussian Naïve Bayes, etc. The study encoded the data through column transformer and one hot encoding before using count vectorizer in the data.

NAÏVE BAYES CLASSIFIER

As for Naïve Bayes Classifiers, (Kumar, 2023) stated that it is a collection of classification algorithms that is based on Bayes' Theorem. The common rule among these algorithms is that each classified feature is independent from each other.

(Hadi Al Ghozali, Pirman, & Indra, 2023)'s research explored the hate speech classification in Indonesian social media. Their hate speech was mainly classifier into six different categories, which are insults, unpleasant acts, fake news,

inciting, provoking, and defamation. The authors made a comparative study on SVM and Naïve Bayes' performance, using Indonesian NER (InNER) as a method to identify hate speech.

(Omran, Al Tararwah, & Al Qundus, 2023) studied Twitter text data for hate speech detection. They performed a comprehensive comparative study on the performance of different machine learning algorithms. Their results show that Naïve Bayes with Decision Tree gave the highest performance for detecting hate speech. The researchers noted that the methodology implemented utilized the strengths and limitations of each algorithm, helping enhance the understanding in hate speech detection.

DATASET AND COLLECTION

The dataset was collected from Kaggle. Tweeter was the only source of the dataset, collecting tweets that are either cyberbullying or not cyberbullying. Furthermore, the cyberbullying tweets were classified into 5 other categories, namely Age, Ethnicity, Gender, Religion, and Other Types. The dataset contains 47,000 labelled tweets, giving balanced data that has around 8000 tweets per class.

Given the nature of the dataset, opportunities for both binary classification and multiclass classification are available for the student to study.

Kaggle Dataset: https://www.kaggle.com/datasets/andrewmvd/cyberbullying-classification

BINARY CLASSIFICATION

This section discusses the end-to-end process of binary hate speech classification. The student uses three models, Random Forest, Logistic Regression, and Naïve Bayes, on the dataset and compares their results. The discussion will address patterns and findings seen in the models' results as well as identifying the best model for binary classification.

DATA PRE-PROCESSING

For data pre-processing, the student first encoded the cyberbullying_type column, which contains six different classes, into 1 or 0. 1 is for hate speech, while 0 represents not hate speech. Then, they randomly sampled the rows categorized as hate speech, the number of samples equaling the number of not hate speech labels. The reason behind the random sampling is due to the large disparity between hate and non-hate data, 39,747 and 7,945 respectively.

The text pre-processing started with a removal of emojis and emoticons, followed by a standardization of lower case text. Twitter handles, punctuations, numbers, URLs, and stop words were removed. Stop words are frequently used words, like “a,” “the,” etc., which add little to no useful information. Moreover, the data size also decreases with the removal of these stop words. Lastly, stemming was performed on the dataset. According to (Sharma, 2022), stemming is an NLP technique that changes the word into its root form.

From here, two stems of pre-processing were done. The first did not remove outliers. The second version removed outliers in the 5th percentile and beyond the 95th percentile. After evaluating both runs, the second version gave a better performance.

MODEL DEVELOPMENT

The student used pipelines for their models, Random Forest, Logistic Regression, and Naïve Bayes. The Random Forest and Logistic Regression pipelines included a TF-IDF Vectorizer and a Truncated SVD step before going into the classifiers. For Naïve Bayes, the pipeline has skipped the Truncated SVD step since the algorithm performs well for high dimension data.

According to (Chaudhary, 2020), TF-IDF means Term Frequency Inverse Document Frequency. Like the name suggests, it represents the text in numbers, which is then used for algorithms. It computes the numerical values by taking the frequency of the term divided by the total number of terms in the text. Then, the quotient would then be multiplied by the inverse document frequency. As for Truncated SVD, it is a transformer that reduces dimensionality in the data (Scikit-Learn, n.d.).

RESULTS

RF Classification Report:

	precision	recall	f1-score	support
0	0.82	0.87	0.84	1339
1	0.85	0.79	0.82	1268
accuracy			0.83	2607

Figure 1 Binary Classification Random Forest Results

The random forest classification results show decent scores across precision, recall, F1-score, and accuracy. An overall accuracy of 0.83 was seen from random forest.

Logistic Regression Classification Report:

	precision	recall	f1-score	support
0	0.80	0.88	0.84	1339
1	0.86	0.76	0.81	1268
accuracy			0.83	2607

Figure 2 Binary Classification Logistic Regression Results

As for logistic regression, the model also shows good scores across all metrics. It has an overall accuracy of 0.83.

Naive Bayes Classification Report:

	precision	recall	f1-score	support
0	0.82	0.71	0.76	1339
1	0.73	0.84	0.78	1268
accuracy			0.77	2607

Figure 3 Binary Classification Naive Bayes Results

Lastly, for Naïve Bayes, the model performed the worst compared to the other two models since it has lower in F1-scores. The overall accuracy of the model is 0.77, which is the lowest among the three.

DISCUSSION

Based on the models' results, logistic regression and random forest have the best performances, but depending on the user, they may choose one of the three models. If recall, or the portion of correctly identified hate speech, then Naïve Bayes may be the best since it scored the highest recall for hate speech among the three models. The model does have a lower accuracy compared to the others. However, in this situation, the consequences of not detecting hate speech are more severe than incorrectly classifying non-hate speech as hate speech.

MULTICLASS CLASSIFICATION

In this section, the student evaluated models on their performance to classify data into six different classes, namely Not Bullying (0), Gender (1), Religion (2), Other Types of Cyberbullying (3), Age (4), and Ethnicity (5). The student will go through the pre-processing steps made, model pipeline, and results of the models. A discussion is made about the best model and about the common findings about the three models.

DATA PRE-PROCESSING

Like in the binary classification, the multiclass classification's pre-processing removed emoticons and cleaned the text by removing twitter handles, changing text to lower case, removing punctuations, numbers, punctuations, and stop words, and lastly, stemming. Vectorization was not included in pre-processing since it is integrated into the model pipeline.

MODEL DEVELOPMENT

For the models, pipelines were used from the sklearn library for uniformity and easier tracking among all models. Only three multiclass classification algorithms were used here, like binary classification, which are Random Forest Classifier, Logistic Regression, and Naïve Bayes Classifier.

All pipelines start with a TF-IDF Vectorizer, which converts the text into numerical format. A maximum of 5000 features was set for the TF-IDF Vectorizer to prioritize more informative features in the data. For random forest and logistic regressions, a Truncated SVD step was included to reduce the dimensionality of the data. However, for Naïve Bayes, since the model performs well with high dimension data, the Truncated SVD step was not included. Lastly, the data was fit into the classifiers.

RESULTS

This part will discuss the results of Random Forest, Logistic Regression, and Naïve Bayes Classifiers on multiclass data. Four metrics are considered when evaluating the models, which are precision, recall, F1-score, and accuracy.

RF Classification Report:

	precision	recall	f1-score	support
0	0.93	0.97	0.95	1603
1	0.96	0.95	0.95	1603
2	0.85	0.79	0.82	1531
3	0.51	0.45	0.48	1624
4	0.48	0.55	0.51	1612
5	0.92	0.92	0.92	1566
accuracy			0.77	9539

Figure 4 Multiclass Classification Random Forest Results

The random forest shows good precision, recall, and F1-score for not-cyberbullying (0), gender (1), and ethnicity (5), scoring more than 90% in the metric. Meanwhile, religion (2) performed in the midline when compared to the other classes. However, other types of cyberbullying (3) and age (4) scored poorly,

specifically in the precision and recall metrics. These lower scores show that the model has difficulty predicting for these classes. Overall, the accuracy of the random forest model is 0.77, which can indicate some room for improvement.

Logistic Regression Classification Report:

	precision	recall	f1-score	support
0	0.92	0.95	0.94	1603
1	0.97	0.95	0.96	1603
2	0.88	0.76	0.82	1531
3	0.50	0.50	0.50	1624
4	0.54	0.61	0.57	1612
5	0.93	0.91	0.92	1566
accuracy			0.78	9539

Figure 5 Multiclass Classification Logistic Regression Results

The logistic regression model shows the same results for non-cyberbullying (0), gender (1), and ethnicity (5) related tweets. Religion (2) is still in the middle, while other types of cyberbullying (3) and age (4) show slightly better performance in the logistic regression model. Logistic regression's accuracy is slightly above random forest's accuracy at 0.78.

Naive Bayes Classification Report:

	precision	recall	f1-score	support
0	0.78	0.95	0.86	1603
1	0.85	0.88	0.87	1603
2	0.75	0.80	0.77	1531
3	0.64	0.41	0.50	1624
4	0.59	0.51	0.55	1612
5	0.80	0.96	0.87	1566
accuracy			0.75	9539

Figure 6 Multiclass Classification Naive Bayes Results

The Naïve Bayes Classifier performed the worst among the three models. Although its precision for the difficult class, other types of cyberbullying (3) and age (4), are the best compared to random forest and logistic regression, the other

classes performed poorly. Overall, the accuracy of Naïve Bayes is 0.75, the lowest among the three models.

DISCUSSION

Based on the models' performances, the best model here is logistic regression since it had the highest in precision, recall, and F1-score with classes 0, 1, and 5. The data of difficult classes, other types of cyberbullying (3) and age (4), may have been more difficult for the models to process. For other types of cyberbullying, may have too much variation in its text data for the vectorizer to pick up anything substantial for the category. The same goes for age. Since age groups have many variations and there are many stereotypes against different age groups, the data may have been too varied for the models to detect any special

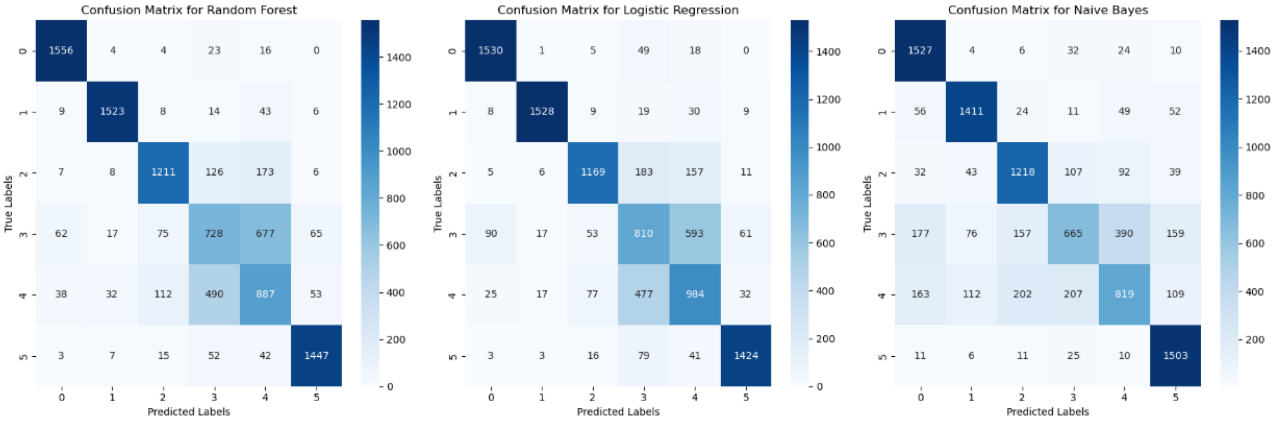

Figure 7 Multiclass Classification Confusion Matrix

words.

In the plotted graphs above, show that random forest and logistic regression had difficulty distinguishing between other types of cyberbullying and age-related cyberbullying tweets. Naïve Bayes on the other hand, shows that the model has more difficulty in distinguishing other types of cyberbullying and age tweets into the right label.

With the three models showing similarities in difficulty for class 3 and 4, better pre-processing practices may need to be implemented for this dataset. Although class 3 and 4 may have more varied text compared to the other classes, other likely culprits of these classes underperforming may have been due to the removal of stop words and short tokens.

CONCLUSION

Hate speech detection remains an interesting and useful topic among data scientists. The project not only helps data scientists explore the pre-processing techniques involved for text data but also deepen the student's understanding of the strengths and weaknesses of different machine learning classification algorithms. In this project, the student was able to learn the importance of other metrics and to not only rely on accuracy as a basis for their judgement. In this hate speech detection, the consequences of not identifying hate speech are far graver than incorrectly labelling non-hate speech as hate speech.

REFERENCES

Anjum, & Katarya, R. (2023). HateDetector: Multilingual technique for the analysis and detection of online hate speech in social networks. *Multimed Tools Appl*.

Avanthika, K., Mrithula, K., & Thenmozhi, D. (2023). SSN-NLP-ACE@Multimodal Hate Speech Event Detection 2023: Detection of Hate Speech and Targets using Logistic Regression and SVM. *Proceedings of the 6th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text*, 66-70.

Chaudhary, M. (2020, April 24). *TF-IDF Vectorizer scikit-learn*. Retrieved from Medium: https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a

Das, S., Bhattacharyya, K., & Sarkar, S. (2023). Performance Analysis of Logistic Regression, Naive Bayes, KNN, Decision Tree, Random Forest and SVM on Hate Speech Detection from Twitter. *International Research Journal of Innovations in Engineering and Technology (IRJIET)*, Vol 7, Issue 3, 24-28.

Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. O'Reilly Media, Inc.

Hadi Al Ghozali, I., Pirman, A., & Indra, I. (2023). Comparison of SVM and Naïve Bayes Algorithms with InNER enriched to Predict Hate Speech. *Makaleler*, Vol 10 Issue 3 600-611.

Kumar, N. (2023, September 13). *Naive Bayes Classifiers*. Retrieved from Geeks for Geeks: https://www.geeksforgeeks.org/naive-bayes-classifiers/

Ndenga, K. (2023). A DEEP DECISION FORESTS MODEL FOR HATE SPEECH DETECTION. *Jordanian Journal of Computers and Information Technology (JJCIT)*, Vol. 09, No. 01 53-62.

Omran, E., Al Tararwah, E., & Al Qundus, J. (2023). A comparative analysis of machine learning algorithms for hate speech detection in social media. *Online Journal of Communication and Media Technologies*, Vol 13, Issue 4.

Safdar, K., Nisar, S., Iqbal, W., Ahmad, A., & Bangash, Y. (2023). Demographical Based Sentiment Analysis for Detection of Hate Speech Tweets for Low Resource Language. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Sciki-Learn. (n.d.). *sklearn.decomposition.TruncatedSVD¶*. Retrieved from Scikit-Learn: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html

Sharma, P. (2022, September 1). *An Introduction to Stemming in Natural Language Processing*. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2021/11/an-introduction-to-stemming-in-natural-language-processing/