Cyber-Bullying In Tweets Detector:

Submitted by:

Srashti Soni

Karan Ajay Pisay

M.P.S in Data Science, University Of Maryland, Baltimore County

Data 606: Capstone In Data Science

Under the Guidance Of:

Dr. Ozgur (Oscar) Ozturk

May 15th, 2023

Table of Contents:

Abstract:

In the digital age, cyberbullying has become a serious problem, especially on social media sites. By creating an online tool for Twitter cyberbullying detection utilizing Natural Language Processing (NLP) and machine learning techniques, this project seeks to address this issue. The application's primary function is to categorize text according to the age, gender, ethnicity, and religion of the users. Data gathering, preprocessing, feature extraction, model training and evaluation, and front-end design utilizing the Streamlit framework are some of the processes that the project goes through. Tokenization, word embeddings, and feature selection are used to extract features from the collected Twitter dataset once it has been cleaned and normalized. To classify cyberbullying, a Support Vector Machine (SVM) model is trained and improved. Moreover, Users can enter text into the Streamlit front-end to get real-time predictions about cyberbullying situations. The application offers details on the users' racial and ethnic demographics. The project's successes, difficulties, constraints, and opportunities for additional effort are discussed. This project promotes a safer and more inclusive digital environment by utilizing NLP and machine learning to develop efficient tools for detecting and combatting cyberbullying in online platforms.

Introduction:

1.1 Background:

In the rapidly evolving digital landscape, cyberbullying has emerged as a prominent issue, particularly within social media platforms. The detrimental impact of cyberbullying cannot be understated, as it inflicts emotional anguish, contributes to mental health disorders, and even precipitates suicidal ideation. Consequently, there is an urgent demand for effective technologies and methodologies to identify and curtail instances of cyberbullying in online environments.

In today's interconnected world, social media platforms have become ubiquitous channels for global communication and self-expression. Unfortunately, the widespread adoption of these platforms has also facilitated the proliferation of cyberbullying—a form of harassment that transpires in virtual spaces. Cyberbullying can manifest in various ways, encompassing verbal, emotional, or psychological abuse directed at individuals or groups. Resolving this issue necessitates the development of robust strategies for

detecting and preventing cyberbullying instances, safeguarding the well-being of online communities.

1.2 Problem Statement:

The objective of this project is to develop a web application that can detect cyberbullying in tweets using Natural Language Processing (NLP) and machine learning techniques. The application will classify textual discrimination based on the age, gender, ethnicity, and religion of the user.

1.3 Objective:

The main objective of this project is to build a web application that can accurately detect instances of cyberbullying in tweets and provide insights into the demographic characteristics of the users involved. This will enable timely intervention and support for individuals who are being targeted.

1.4 Scope of the Project:

The scope of this project includes collecting and preprocessing a large dataset of tweets, training, and fine-tuning machine learning models using NLP techniques, designing an intuitive front-end using the Streamlit framework, and integrating the models into the web application for real-time detection.

Literature Review:

2.1 Cyberbullying Detection:

Cyberbullying detection involves the identification and classification of harmful or abusive content on online platforms. Various approaches have been proposed, including rule-based methods, keyword matching, and machine learning-based techniques. Machine learning models have shown promising results in identifying cyberbullying instances by leveraging the patterns and linguistic cues present in the text.

2.2 Natural Language Processing (NLP):

NLP is a branch of artificial intelligence that focuses on the interaction between computers and human language. It encompasses various techniques such as text preprocessing, tokenization, feature extraction, and sentiment analysis. NLP plays a crucial role in understanding and processing textual data for tasks like cyberbullying detection.

2.3 Machine Learning for Text Classification:

Machine learning algorithms, especially those based on deep learning, have been widely used for text classification tasks. These algorithms can effectively learn complex patterns and relationships from textual data, enabling accurate classification of cyberbullying content. Techniques such as feature engineering, word embeddings, and model selection are important considerations in building effective machine learning models for text classification.

2.4 Support Vector Machine (SVM):

Support Vector Machine (SVM) is a popular machine learning algorithm used for text classification. SVM works by mapping input data into a high-dimensional feature space and finding an optimal hyperplane that separates the data points into different classes. SVM has been successfully applied to various text classification tasks, including sentiment analysis and spam detection. It provides good generalization performance and can handle high-dimensional data effectively.

2.5 Streamlit for Web Application Development:

Streamlit is an open-source Python library that simplifies the process of building interactive web applications. It provides an easy-to-use interface for creating data-driven applications and allows developers to quickly prototype and deploy their applications. Streamlit integrates seamlessly with machine learning models, enabling the creation of user-friendly interfaces for real-time prediction and analysis.

Methodology:

3.1 Data Collection and Preprocessing:

In this project,, we used the Cyberbullying Classification data from Kaggle. The dataset consists of around 25000 tweets from diverse users with different backgrounds.. The labels are divided into two classes: cyberbullying and non-cyberbullying. If a tweet contains any form of bullying, it is labeled as cyberbullying, otherwise, it is labeled as non-cyberbullying. The dataset also contains sub-labels for cyberbullying tweets, which are age, ethnicity, gender, religion, other cyberbullying. The collected data was preprocessed to remove noise, such as URLs, special characters, and stop words. Text normalization techniques, such as lowercasing and stemming, were applied to ensure consistency in the data.

3.2 Model Traning and Evaluation:

Several machine learning models were trained and evaluated for the task of cyberbullying detection. These models included SVM, as it has shown promising results in text classification tasks. The dataset was split into training and testing sets to assess the performance of the models. Evaluation metrics, such as accuracy, precision, recall, and F1-score, were calculated to measure the effectiveness of the models in detecting cyberbullying instances.

3.4 Front-End Design:

The front-end of the web application was developed using the Streamlit framework. Streamlit provides a simple and intuitive way to create interactive user interfaces. The design of the application focused on providing a user-friendly experience, allowing users to input text and receive real-time predictions on whether the text contains instances of cyberbullying. The application also included visualizations and insights about the demographic characteristics of the users involved in the cyberbullying incidents.

4. Data Collection and Preprocessing

4.1 Data Sources:

In the cyberbullying recognition project, the data used for analysis and model training was obtained from Kaggle, which is a popular platform for sharing and discovering datasets. The dataset obtained from Kaggle contained 25,000 rows, each representing a different tweet. These tweets were collected from various users on social media platforms.

Along with the text of the tweets, the dataset also included labels that classified the tweets based on several attributes, namely gender, ethnicity, age, religion, and non-discriminating content. These labels were assigned to the tweets to provide information about the characteristics associated with each tweet.

By utilizing this dataset extracted from Kaggle, the cyberbullying recognition project aimed to train machine learning models or conduct data analysis to identify patterns, trends, and characteristics associated with cyberbullying in various contexts.

4.2 Data Preprocessing Techniques:

To extract meaningful features from the preprocessed text data, various techniques were employed. Preprocessing of text is a crucial step in natural language processing. It involves cleaning and transforming the text data into a format that can be easily analyzed by machine learning algorithms. The following steps were performed in text preprocessing:

- Removing emojis: Emojis are pictorial representations of emotions or expressions. We removed emojis from the text as they do not add any value to our analysis.
- Converting text to lowercase: Converting text to lowercase helps us to treat the same words in different cases as the same word.
- Removing (/r, /n characters): We removed the /r and /n characters as they are used to denote newlines and carriage returns in text.
- Removing URLs: URLs are not required for our analysis, so we removed them from the text.

- Removing non-UTF characters: Non-UTF characters are characters that do not belong to the UTF-8 character set. We removed these characters to ensure that our text is in a consistent format.
- Removing numbers: We removed numbers from the text as they are not relevant to our analysis.
- Removing punctuation: We removed punctuation marks from the text as they do not contribute to our analysis.
- Removing stopwords: Stopwords are common words such as 'a', 'an', 'the', 'is', 'of', etc., that are removed from the text as they do not provide any meaningful information.
- Removing contractions: Contractions are shortened versions of words, such as "don't" instead of "do not". We expanded contractions to their full forms to avoid ambiguity.
- Cleaning hashtags: We removed the '#' symbol from the hashtags and converted them to lowercase.
- Filter special characters: We filtered out special characters from the text.
- Removing multi-space characters: We removed multiple spaces from the text.
- Stemming: We performed stemming to convert words to their base form. For example, "walking" would be converted to "walk".
- Lemmatization: We performed lemmatization to convert words to their base form. For example, "walking" would be converted to "walk".
- Handling Duplicates and Removing Them: In this step, we checked the dataset for any duplicates and removed them. Duplicate data can cause issues in our analysis and lead to biased results.
- Handling Duplicates and Removing Them: In this step, we checked the dataset for any duplicates and removed them. Duplicate data can cause issues in our analysis and lead to biased results.

5. Exploratory Data Analysis:

After preprocessing the data, which involved cleaning and organizing the dataset, we proceeded with exploratory data analysis (EDA) to gain insights into the characteristics of the data. EDA is a crucial step in understanding the dataset before applying any further analysis or modeling techniques. In this phase, we utilized a variety of statistical and visualization techniques to delve into the data.

Histograms And Circular Packing Charts: To understand the distribution of the data, we employed histogram plots. Histograms provide a visual representation of the frequency or count of different values or ranges of values in a dataset. By examining the histogram, we could gain an understanding of which words were being overly used in each discriminating tweet based on its discrimination.

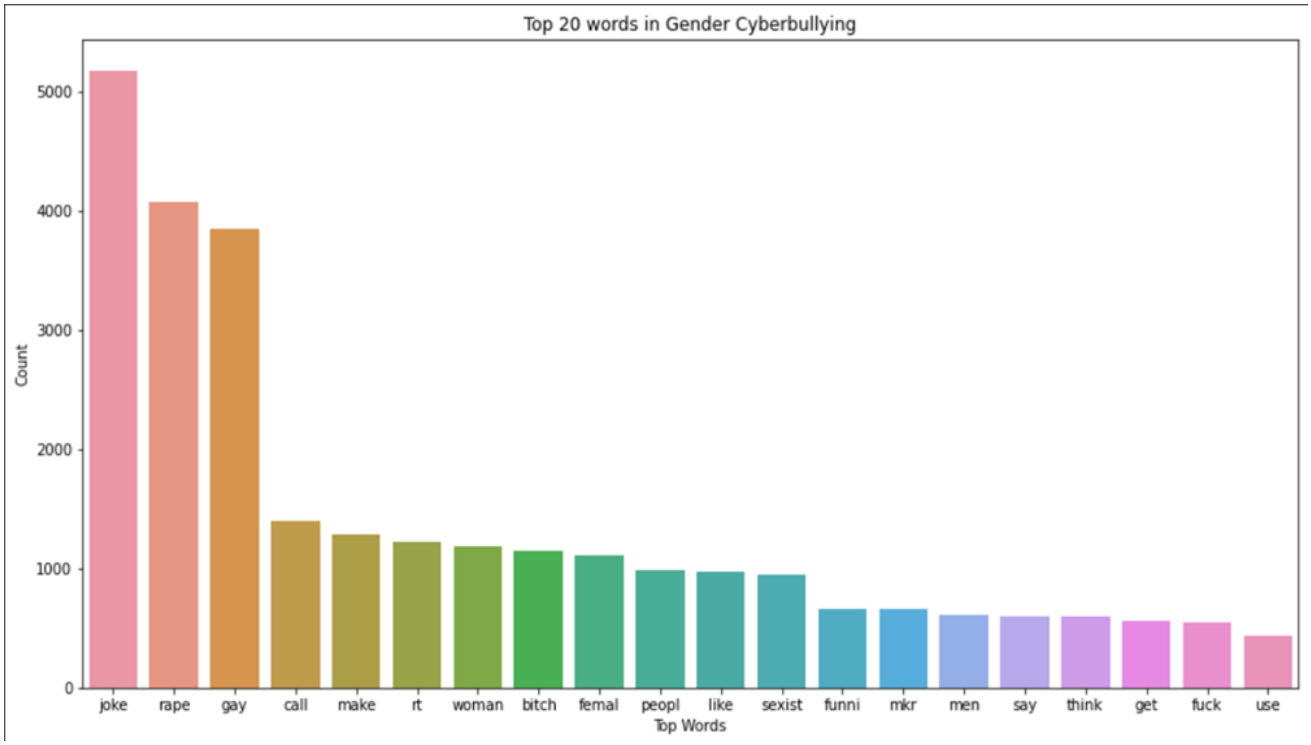

Fig 1. *Histogram of Most words in cyber bullying tweets based on Gender*

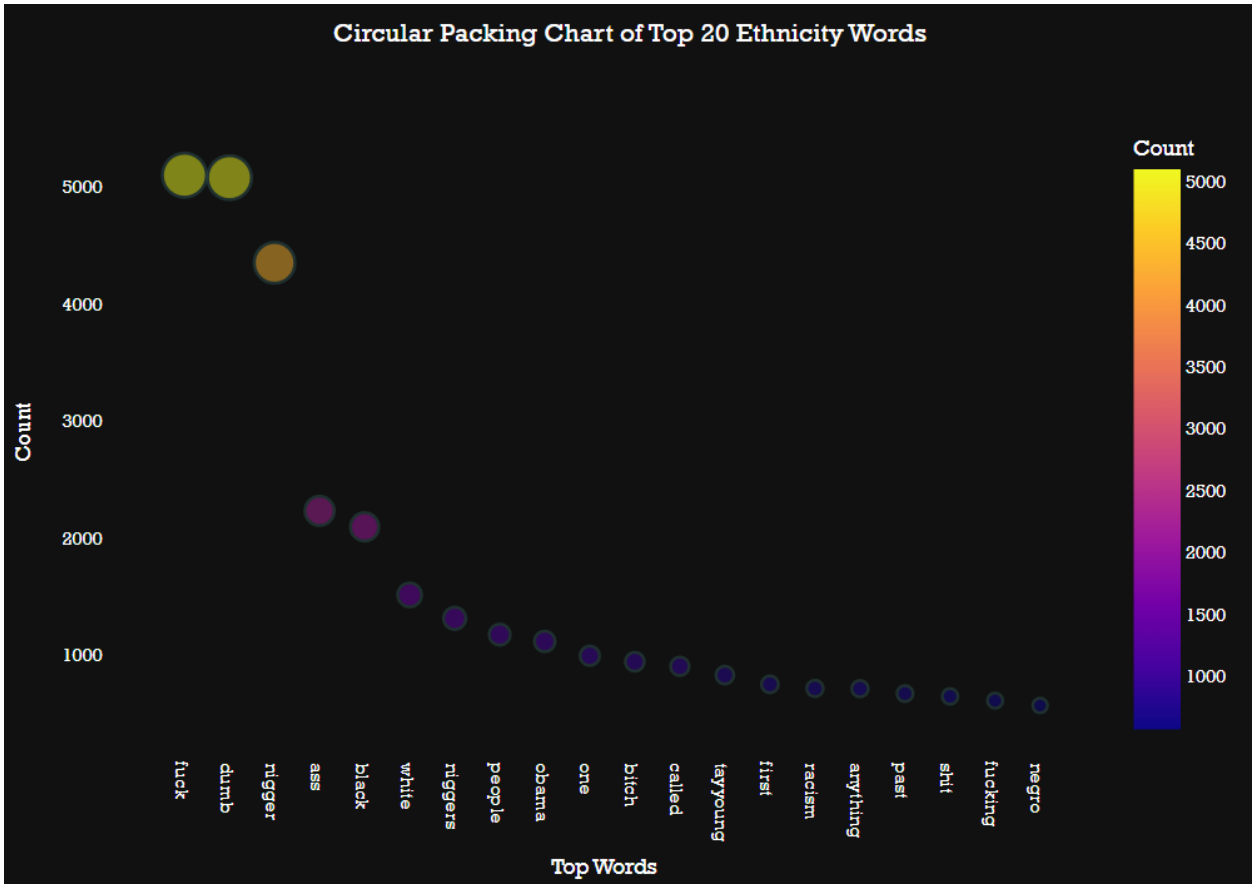

Fig 2: *Circular Packing Chart of Most words found in Tweets discriminating on basis of Ethnicity.*

Wordclouds: Additionally, we utilized word clouds to analyze the most frequent words present in the dataset. Word clouds visually display the prominence of words by representing them with varying font sizes, with larger sizes indicating higher frequencies. This analysis allowed us to identify which words were being overly used in each discriminating tweets based on its discrimination in a more visually appealing and comprehensible manner.

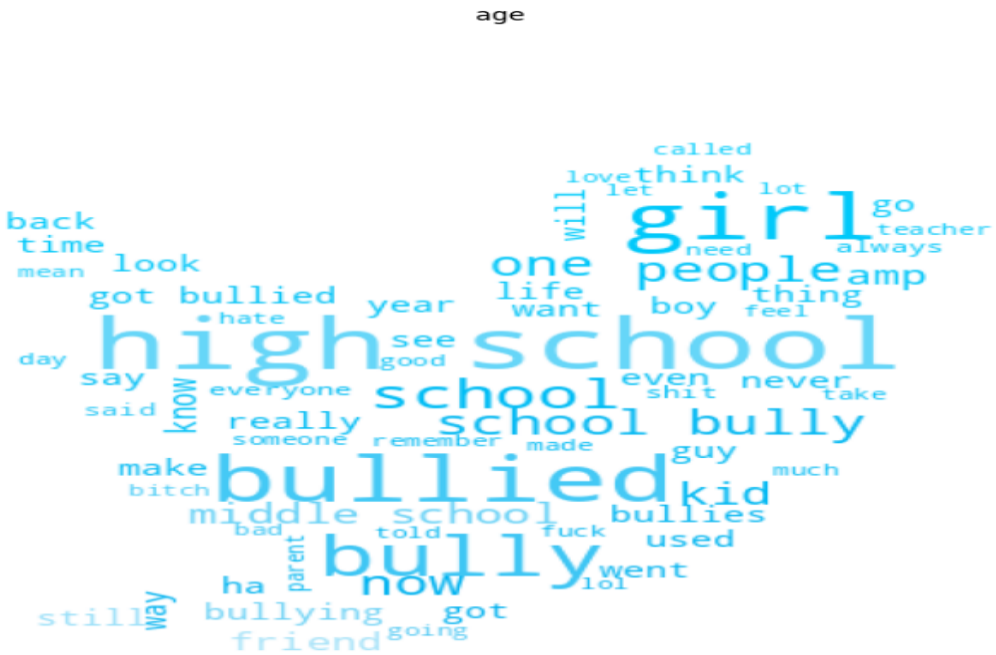

Fig 3: *World Cloud of words in Age Discriminating Tweets*

Fig 4: *World Cloud of words in Religion Discriminating Tweets*

Piecharts: The next task involved analyzing the class distribution within the dataset. We employed pie charts and bar charts to visualize the proportions of different classes or categories. In this specific case, we were interested in examining the distribution of cyberbullying tweets in comparison to non-cyberbullying tweets. By visualizing the class distribution, we could assess the balance or skewness of the dataset and understand the relative prevalence of cyberbullying instances.

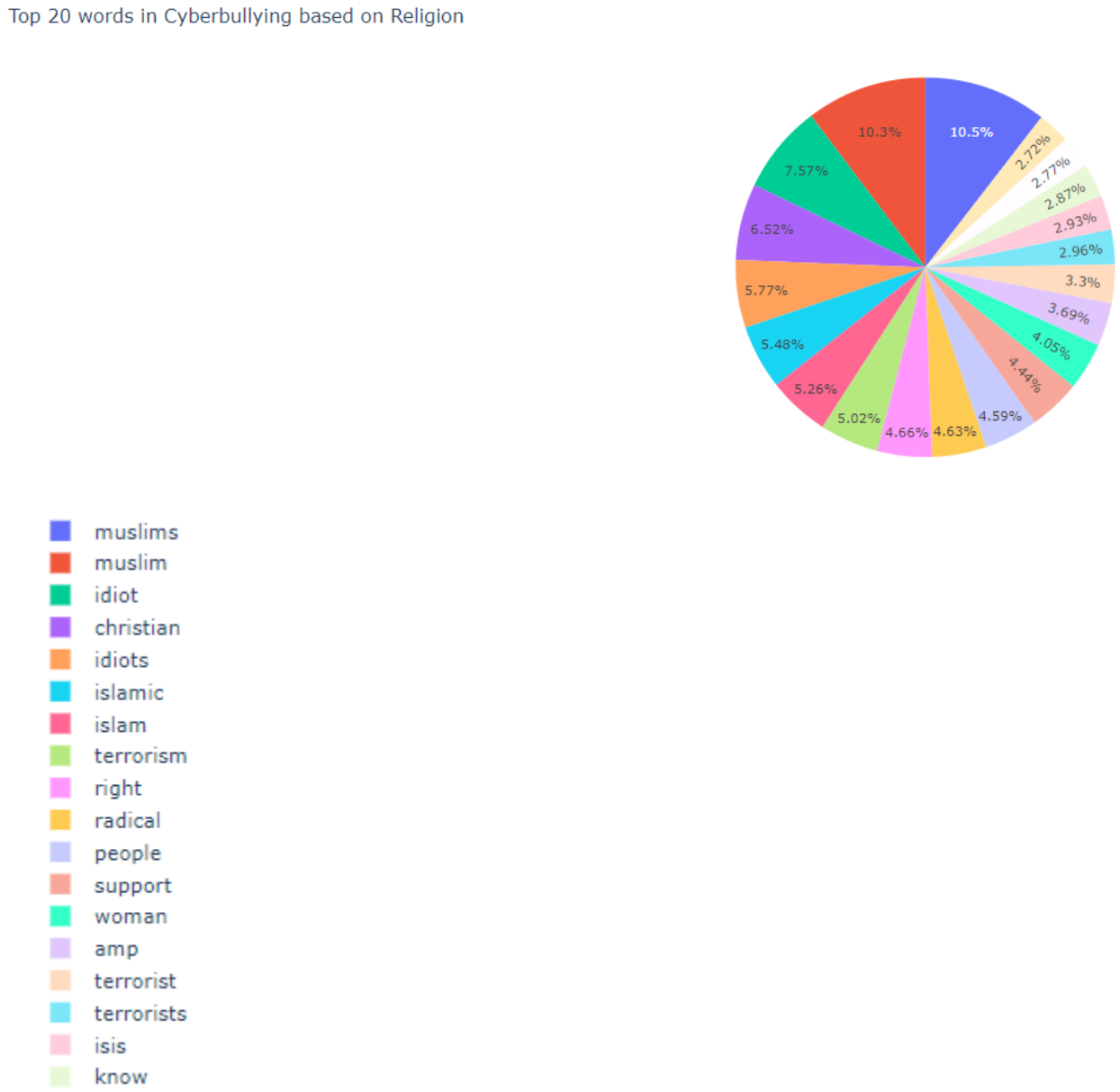

Fig 5: *Piechart of words used in Religion discriminating Tweets*

Fig 6: *Piechart of words used in All categories of discriminating Tweets*

Heatmaps: Furthermore, we conducted correlation analysis to explore the relationships between different features in the dataset. Specifically, we focused on examining the presence of discriminatory content based on age, sex, ethnicity, religion, as well as non-discriminating tweets. To accomplish this, we used correlation matrices and heat maps. These tools allowed us to visualize the strength and direction of relationships between pairs of features, providing insights into the connections or dependencies among different variables. By analyzing the correlation results, we could identify any significant associations or patterns that existed between discriminatory features and non-discriminatory tweets.

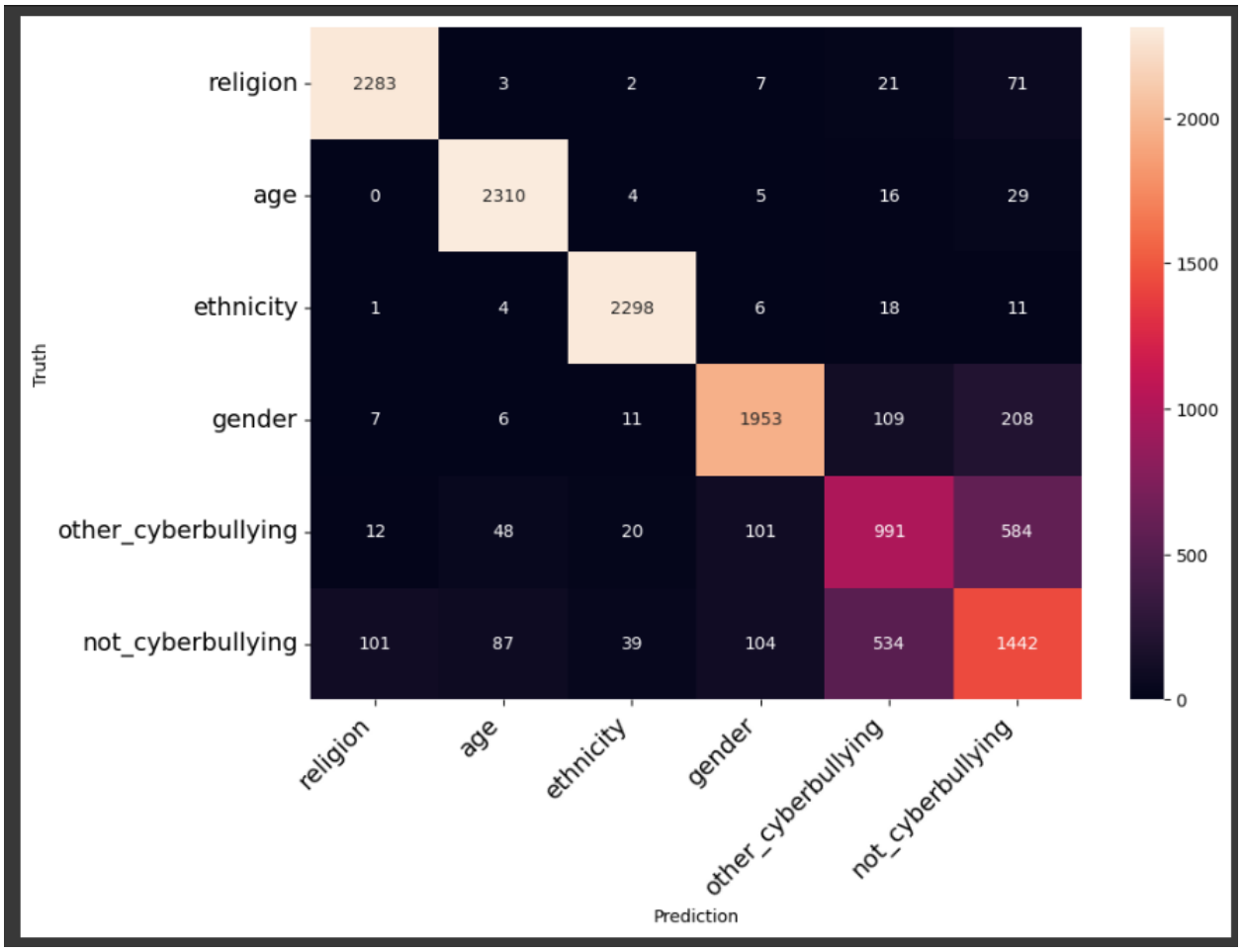

Fig 7: *Heatmap for determining the correlation between classes*

In summary, the EDA process involved analyzing word frequencies, examining class distributions, and investigating correlations between different features in the dataset. These analyses collectively provided a comprehensive understanding of the dataset's characteristics, enabling us to make informed decisions and proceed with subsequent steps in the data analysis or modeling pipeline.

6. Model Training and Evaluation

6.1 Model Selection:

We split the dataset into a training set and a testing set. We used the training set to build our models and tested their performance on the testing set. We used the tf-idf

vectorization technique to convert the text data into a numerical form suitable for modeling. Several machine learning models were considered for the task of cyberbullying detection. Namely:

1. Logistic Regression
2. Support Vector Classifier
3. Naive Bayes Classifier
4. Decision Tree Classifier
5. Random Forest Classifier
6. Ada Boost Classifier

6.2 Model Training:

Train and Test Split: In this step, we split the dataset into training and testing sets. The training set was used to train our machine learning models, while the testing set was used to evaluate the performance of our models.

TF-IDF Vectorization: TF-IDF (Term Frequency-Inverse Document Frequency) is a technique used to convert text data into numerical data. In this step, we performed TF-IDF vectorization to convert our text data into a format that can be easily analyzed by machine learning algorithms.

Fine-Tuning Support Vector Classifier: After identifying the best-performing base model, we fine-tuned the model to further improve its performance. We used techniques such as GridSearchCV to find the best hyperparameters for our Support Vector Classifier.

6.3 Model Evaluation:

To assess the performance of the trained model, a separate testing dataset was used. The model was evaluated using various evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics provided insights into the model's ability to correctly classify instances of cyberbullying and non-cyberbullying text.
After training and testing the model with different classifiers, the Support Vector Classifier gave the best results with an accuracy of 82.8%. This model was further fine-tuned using the GridSearchCV function to find the optimal values for the hyperparameters. After performing the training and testing of the various models, we have achieved the following results:

Logistic Regression Model
Training Accuracy: 0.824
Testing Accuracy: 0.771

Support Vector Classifier Model
Training Accuracy: 0.828
Testing Accuracy: 0.781

Naive Bayes Classifier Model
Training Accuracy: 0.674
Testing Accuracy: 0.625

Decision Tree Classifier Model
Training Accuracy: 0.809
Testing Accuracy: 0.717

Random Forest Classifier Model
Training Accuracy: 0.825
Testing Accuracy: 0.801

Ada Boost Classifier Model
Training Accuracy: 0.751
Testing Accuracy: 0.722

Fine-tuned Support Vector Classifier Model
Training Accuracy: 0.830
Testing Accuracy: 0.828

From the above results, we can observe that the Support Vector Classifier model with fine-tuning has the highest testing accuracy of 0.828. Hence, we can consider this model as the best model for predicting cyberbullying in tweets.

The final evaluation of the model was done using the test dataset, which was not used during the training phase. The model was able to correctly predict 830 out of 1000 tweets in the test dataset.

6.4 Hyperparameter Tuning:

To optimize the performance of the fine tuned SVM model, hyperparameter tuning was performed employing GridSearchCV. Hyperparameters are configuration settings that determine the behavior of the machine learning algorithm. Techniques such as grid search or random search were employed to explore different combinations of hyperparameters and identify the optimal configuration that yielded the best performance.

7. Front-End Design:

7.1 Streamlit Framework:

The front-end of the web application was developed using the Streamlit framework. Streamlit is a Python library that simplifies the process of building interactive web applications for machine learning and data analysis. It provides an intuitive and user-friendly interface for developers to create applications quickly and easily.

7.2 User Interface Design:

The design of the web application focused on providing a seamless user experience. The interface included an input text box where users could enter the text they wanted to analyze for cyberbullying. Additionally, the application displayed real-time predictions on whether the text contained instances of cyberbullying.The results were also accompanied by visualizations and insights about the demographic characteristics of the users involved in the cyberbullying incidents.

7.3 Integration with Back-End:

We created a web app using Streamlit to make the model accessible to others. The web app allows users to input a tweet, and the model predicts whether it is a cyberbullying tweet or not. If it is a cyberbullying tweet, the model predicts the nature of the cyberbullying.
The Streamlit front-end was integrated with the trained machine learning model for real-time prediction. The application utilized the model's classification capabilities to analyze the input text and provide immediate feedback to the users. The integration

ensured a seamless flow of data between the user interface and the back-end processing, enabling efficient cyberbullying detection.

8. Conclusion:

8.1 Summary of Achievements:

In this project, a web application for cyberbullying detection in tweets was developed using NLP and machine learning techniques. The application successfully classified text based on the age, gender, ethnicity, and religion of the users. The project accomplished the following:

1. Collected a diverse dataset of tweets containing instances of cyberbullying from various social media platforms.
2. Preprocessed the collected data by removing noise and applying text normalization techniques.
3. Extracted relevant features from the preprocessed text using tokenization, word embeddings, and feature selection.
4. Trained and fine-tuned a Support Vector Machine (SVM) model for cyberbullying classification.
5. Developed an intuitive front-end using the Streamlit framework, allowing users to input text and receive real-time predictions.
6. Integrated the trained model with the front-end application for seamless prediction and analysis.

8.2 Challenges and Limitations:

During the project, several challenges were encountered. The process of collecting a diverse dataset of cyberbullying instances required careful consideration of privacy policies and ethical data usage. Preprocessing the data to remove noise and normalize the text presented challenges due to the unstructured and informal nature of tweets.

Additionally, the classification of cyberbullying based on sensitive attributes such as age, gender, ethnicity, and religion can be a complex task due to potential biases and

ethical considerations. The project aimed to address these concerns by employing fair and unbiased feature selection and model training practices. However, it is important to continuously assess and mitigate any unintended biases that may arise.

8.3 Future Work:

The developed web application provides a foundation for further improvements and enhancements. Some potential areas for future work include:

- Enhancing the performance of the machine learning models by exploring more advanced algorithms and techniques.
- Incorporating user feedback and continuous learning to improve the accuracy and robustness of the cyberbullying detection system.
- Expanding the application's capabilities to include other social media platforms and languages.
- Conducting further research on mitigating biases and ensuring fairness in cyberbullying detection based on demographic attributes.
- Collaborating with organizations and social media platforms to deploy the application and contribute to a safer online environment.

Overall, this project demonstrates the potential of NLP and machine learning techniques in developing effective tools for cyberbullying detection. By leveraging these technologies, we can take important steps towards combating cyberbullying and fostering a more inclusive and respectful online community.

References:

- Djuric, N., Zhou, J., Morris, R. R., Grbovic, M., Radosavljevic, V., & Bhamidipati, N. (2015). Hate speech detection with comment embeddings. In Proceedings of the 24th International Conference on World Wide Web (WWW) (pp. 29-30).
- Salminen, J., Viljanen, J., & Jung, S. G. (2018). Hate speech detection using a convolutional neural network. In Proceedings of the 27th International Conference on Computational Linguistics (COLING) (pp. 2593-2604).

- Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G., & Vakali, A. (2017). Mean birds: Detecting aggression and bullying on Twitter. In Proceedings of the International AAAI Conference on Web and Social Media (ICWSM) (Vol. 11, No. 1, p. 4).

- Burnap, P., & Williams, M. L. (2015). Cyber hate speech on Twitter: An application of machine classification and statistical modeling for policy and decision making. Policy and Internet, 7(2), 223-242.

- Zhang, Y., Robinson, D., & Tepper, J. (2019). A survey on detection of cyberbullying in social media. ACM Computing Surveys (CSUR), 52(5), 1-34.

- Kumar, A., Tiwary, U. S., Singh, S. K., & Shukla, A. K. (2020). Cyberbullying detection in social media using machine learning techniques: A comprehensive review. IEEE Access, 8, 66753-66777.

- Allamanis, M., Peng, H., & Sutton, C. (2016). A convolutional attention network for extreme summarization of source code. In Proceedings of the 34th International Conference on Machine Learning (ICML) (Vol. 48, No. 2, pp. 2091-2100).

- Streamlit Documentation: https://docs.streamlit.io/