Cyberbullying Tweets Classifier Model using Natural Language Processing

Submitted By

D. Sai Sasikanth

Under the guidance of

S. Ramprakash

December - 2024

ABSTRACT

The increase in social media has come with some new challenges, including the issue of cyberbullying, which has some severe psychological effects on the victims. This project aims at creating a cyberbullying tweet classifier that will be used to identify harmful content and curb cyberbullying. The project uses NLP to process and analyze tweet data to detect abusive or bullying language. The dataset used is a collection of tweets labeled as bullying or non-bullying, hence suitable for supervised machine learning-based classification. Data preprocessing is the most important step in this project. Techniques such as stopword removal and vectorization were used to transform raw text into numerical features suitable for machine learning models. Count Vectorizer and TF-IDF Vectorizer were used to convert the text into feature vectors, capturing the frequency and significance of words in the dataset. Label Encoding was applied to normalize categorical labels, ensuring compatibility with the classification algorithms. Several machine learning models were used and compared to achieve the best classification performance.

These include Naive Bayes Classifier, AdaBoost Classifier, Random Forest Classifier, Logistic Regression, Decision Tree Classifier, Support Vector Classifier (SVC), and K-Nearest Neighbors (KNN). Each model's performance was evaluated using metrics such as accuracy, precision, recall, and F1-score to determine its effectiveness in identifying cyberbullying tweets. Multiple models were integrated for comprehensive analysis of data; insights on the strengths and weaknesses of every approach were achieved. Naive Bayes performed well in text-based features, and ensemble methods like Random Forest and AdaBoost performed great in classification tasks. LR and SVC produced excellent baseline models, and KNN shed light on instance-based learning. This project gives an idea of the importance that could be attached to machine learning and NLP in addressing societal problems, like cyberbullying. Since this classifier automates the identification of abusive content, it therefore contributes to the creation of a safer online environment. Future work could be oriented toward enhancing the model by using advanced techniques, including deep learning or transformer-based models, for further enhancement in accuracy and adaptability across diverse datasets.

TABLE OF CONTENTS

CHAPTER-1

1 INTRODUCTON

1.1 Cyberbullying on Social media:

The digital age has been generally marked by the phenomenon of cyberbullying, defined as the use of electronic communication media to intimidate, harass, or harm others. Different from traditional bullying, cyberbullying occurs in the virtual world, thus making it difficult to identify those perpetuating the actions and reaching them through social media, messaging apps, and online forums. The widespread use of social media has therefore increased the incidence of cyberbullying, and it needs to be of great concern to society. Twitter, Facebook, Instagram, and TikTok are platforms that are an integral part of daily communication in terms of sharing thoughts, opinions, and experiences. At the same time, they give the room for all evil. Twitter is such a place where a brief text published on it in public form can easily be spread due to the nature of this network. Tweets containing derogatory language, threats, or harassment can go viral, significantly impacting the mental health and well-being of victims. Perhaps, one of the characteristics of cyberbullying is that it follows them. In contrast, face-to-face bullying can be placed in specific locations and time slots, whereas cyberbullying content may be accessed and shared repeatedly, thereby leaving a permanent impact. This permanence and reach make cyberbullying particularly pernicious, as victims cannot escape the harassment. The effects of cyberbullying cut across all ages, and the most sensitive to it would be the adolescents because they are more vulnerable during this stage of development. Cyberbullying victims, according to reports, experienced anxiety, depression, and withdrawal from social activities. Moreover, young victims became detached from life due to self-harm and suicidal thoughts. Besides the above effects, cyberbullying creates a toxic online social environment that discourages constructive debate and conversation. What with the scale and gravity of the problem, dealing with cyberbullying has become the priority concern for policymakers, educators, and technologists. Social media companies have introduced reporting systems, moderation algorithms, and other measures, but these efforts are grossly limited by the sheer volume of content produced every day. In response, researchers are turning to artificial intelligence and machine learning to devise automated systems that help in the identification and prevention of cyberbullying. These machines analyze text, images, and video and identify harmful content for removal or further review. This project focuses on detecting cyberbullying using NLP and machine learning techniques, resulting in a proactive means of dealing with this prevalent issue that can create a safer digital environment.

CHAPTER-2

2 METHADOLOGY

2.1 Overview:

This project is designed to formulate an efficient cyberbullying tweet classifier by leveraging NLP and machine learning techniques. The entire process starts with data preprocessing-an essential step that ensures that the input data are cleaned and of the finest quality with no inconsistencies. Raw data for tweets will be cleaned of stopwords-the most commonly occurring words which don't hold much importance in semantic meaning. This reduces noise and enhances the relevance of features that is extracted during the vectorization process. Feature extraction converts text data into a numerical format. The techniques applied here are two, namely, Count Vectorizer and TF-IDF Vectorizer. This captures the frequency of words in the dataset for Count Vectorizer, and on the other hand, for TF-IDF Vectorizer, weights are assigned based on importance, that is, the word frequency within the document and how unique the word is in other documents. Thus, it helps in ensuring meaningful input for models for correct classification. To normalize the target labels, a Label Encoder was used. This converts categorical data into numerical format, making it compatible with machine learning algorithms, which can then process and interpret the data efficiently.

2.2 Proposed Method:

This is the step where various machine learning models are implemented and evaluated. The models used include Naive Bayes Classifier, AdaBoost Classifier, Random Forest Classifier, Logistic Regression, Decision Tree Classifier, Support Vector Classifier (SVC), and K-Nearest Neighbors (KNN). Each of these algorithms was selected for its unique strengths in handling classification tasks. For example, Naive Bayes is particularly suitable for text data due to its probabilistic nature, while ensemble methods such as Random Forest and AdaBoost provide robustness against overfitting. Logistic Regression and SVC serve as reliable baseline models, and KNN offers insights into instance-based learning. Model evaluation was conducted using standard performance metrics such as accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of each model’s ability to correctly identify cyberbullying tweets. Comparing these results allowed for the selection of the most effective algorithm for the given dataset. This methodology therefore portrays a

systematic approach to the problem of detecting cyberbullying on social media, such that by integration of data preprocessing, feature extraction, and robust classification techniques, the project ensures development into a reliable and scalable solution for the mitigation of cyberbullying impact.

CHAPTER – 3

3 CODE EXPLANATION

3.1 Code explanation:

The entire code of this project has been divided into different sections to gradually build a strong cyberbullying tweet classifier. Here follows the description of the components:

1. **Imported Libraries**

The project starts with the importation of necessary libraries in Python to handle any data manipulation, analysis, and visualization, as well as application of machine learning models. The significant libraries include pandas and numpy for data handling, matplotlib and seaborn for visualization, and sklearn for machine learning models and preprocessing tools.

2. Exploratory Data Analysis (EDA)

EDA is performed to understand the dataset's structure, identify patterns, and detect anomalies. Key steps include inspecting the data for missing values, analyzing the distribution of tweets across different cyberbullying categories, and exploring the overall word frequency. This step lays the foundation for effective preprocessing and model building.

3. Visualizing

Visualization techniques are employed to gain insights into the data:

Word Cloud: It displays the most frequently used words in the tweets under each category of cyberbullying.

Unigrams and Bigrams: This highlights single words and word pairs that appear quite frequently, thus helping in determining common patterns.

Heatmap with Confusion Matrix: It gives a visual representation of model performance by displaying correctly as well as wrongly classified instances.

4. Data Preprocessing

This step includes cleaning and transforming the text data for consistency and relevance:

Stopwords removal: removes words that contribute less to the context and therefore includes articles, conjunctions, and prepositions.

Tokenization: the splitting of the text into smaller units of tokens like words or phrases.

Removing digits: it removes the numeric characters from the input.

Removing emojis: this step ensures the cleaning of emojis to have standardized text.

Removing URLs, special characters, and underscores: ensures clean input by removing all the unnecessary components.

Word classification: this is a segregation of words according to their relevance to each type of cyberbullying.

5. Vectorization

Count Vectorizer and TF-IDF Vectorizer are used to transform the text data into numerical feature vectors. This method counts the word frequency and word importance to help machine learning models understand text.

6. Label Encoding

Label Encoding is used to map the categorical labels of the dataset to numerical values so that they are compatible with the machine learning algorithms.

7. Models

Different models were trained and tested based on their performance:

Naive Bayes Classifier: This is a probabilistic model, suitable for classification tasks involving text.

AdaBoost Classifier: Ensemble method that combines weak learners to improve performance.

Random Forest Classifier: Ensemble method using multiple decision trees for robust classification.

Logistic Regression: Linear model for binary classification, extended for multi-class problems.

Decision Tree Classifier: Splits data hierarchically based on feature importance.

Support Vector Classifier (SVC): Finds the hyperplane that maximizes class separation.

K-Nearest Neighbors (KNN): Instance-based learning by finding the closest training examples.

CHAPTER - 4

4 RESULTS AND TESTING

4.1 Output Screenshots, And Results Analysis:

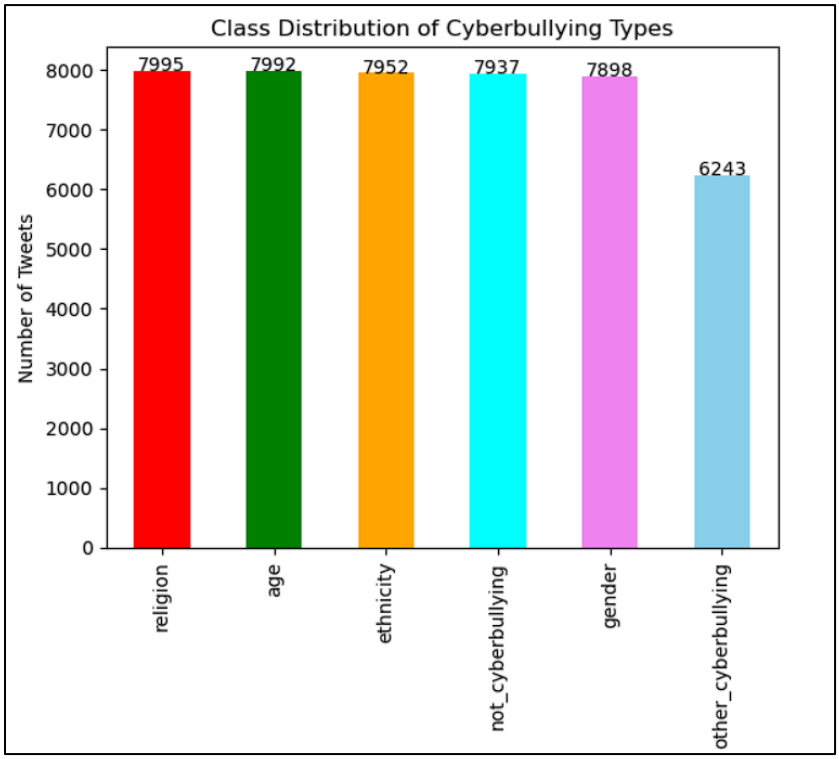
4.1.1 Outputs Explanation:

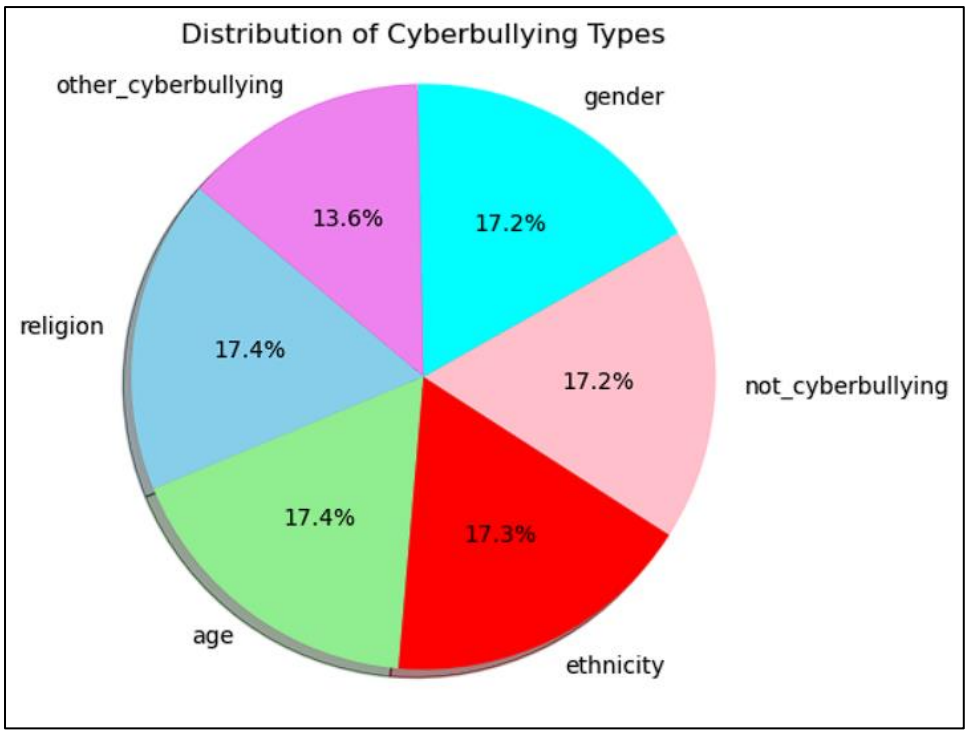
The outputs from the visualizations provide significant insights into the dataset and the performance of the models. Below is a detailed explanation of each visualization:

Class Distribution of Cyberbullying Types (Count and Percentage):

Count Distribution: Shows the frequency of tweets in each cyberbullying type, thus indicating the class imbalance (if any) in the dataset. It would help in identifying the underrepresented classes and informs the need for balancing techniques during model training.

Percentage Distribution: Gives the distribution of each cyberbullying type in percentage value relative to the whole data, giving an overview of how the composition of the data is reflected in percentages.

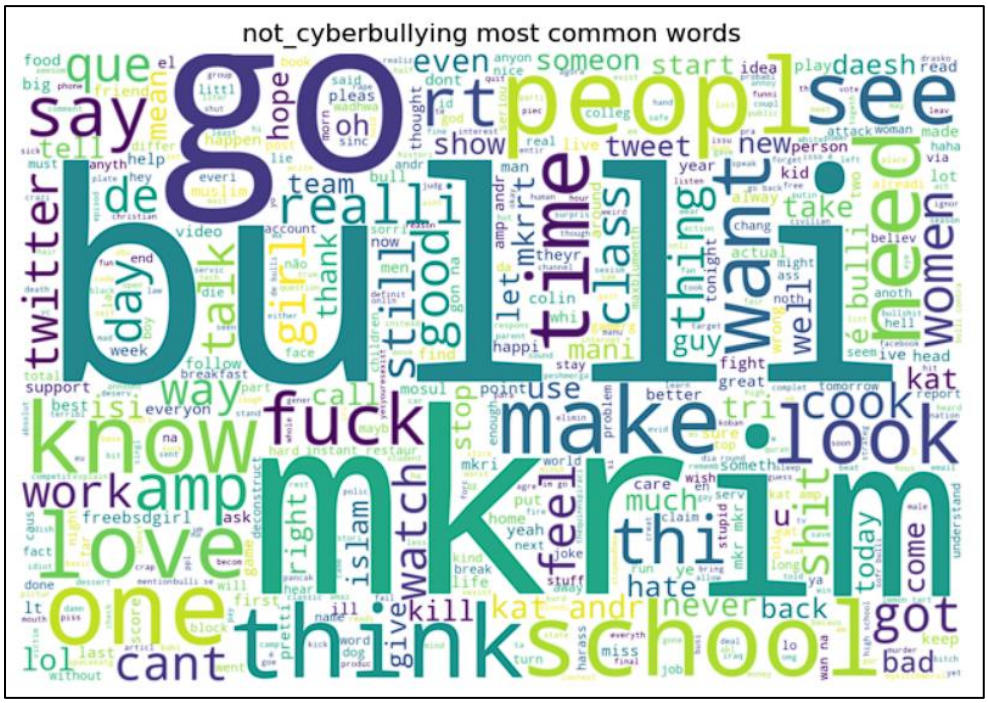


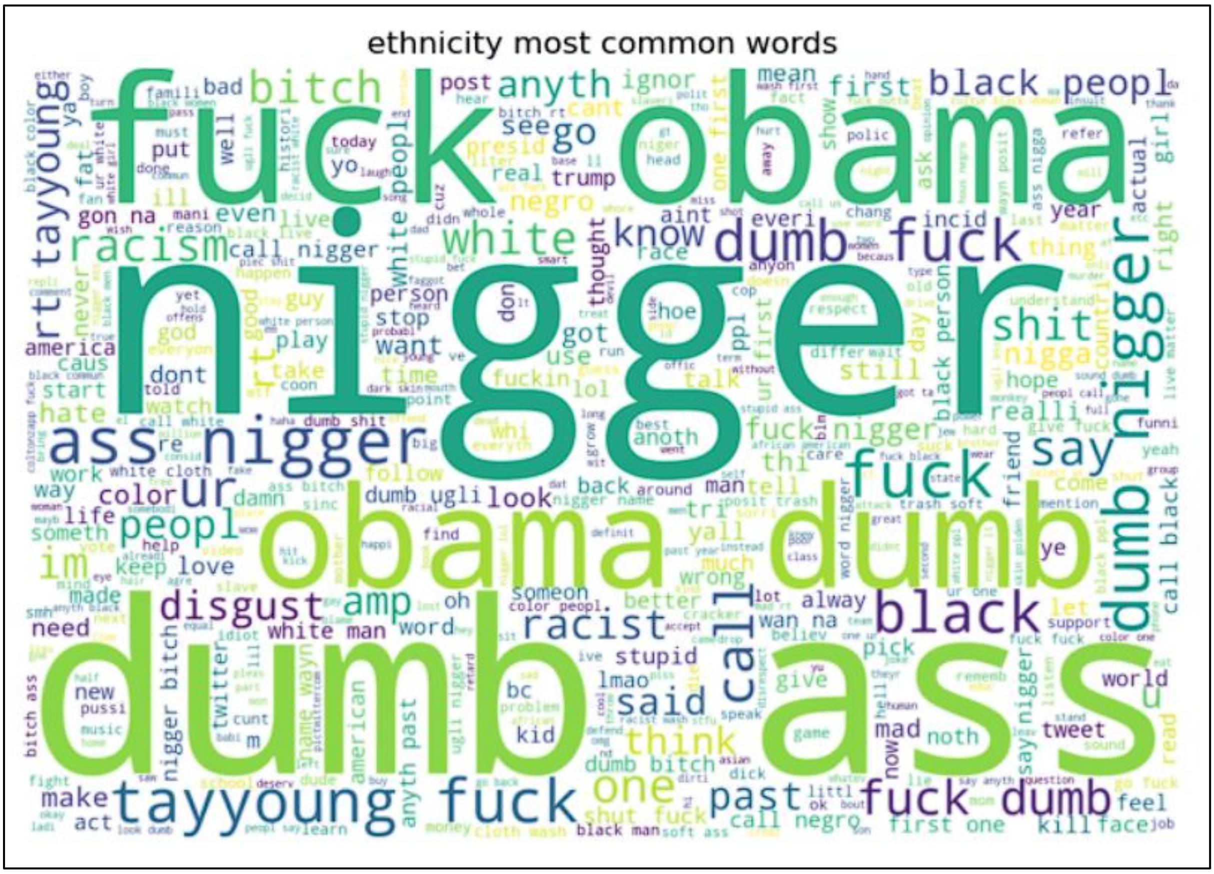


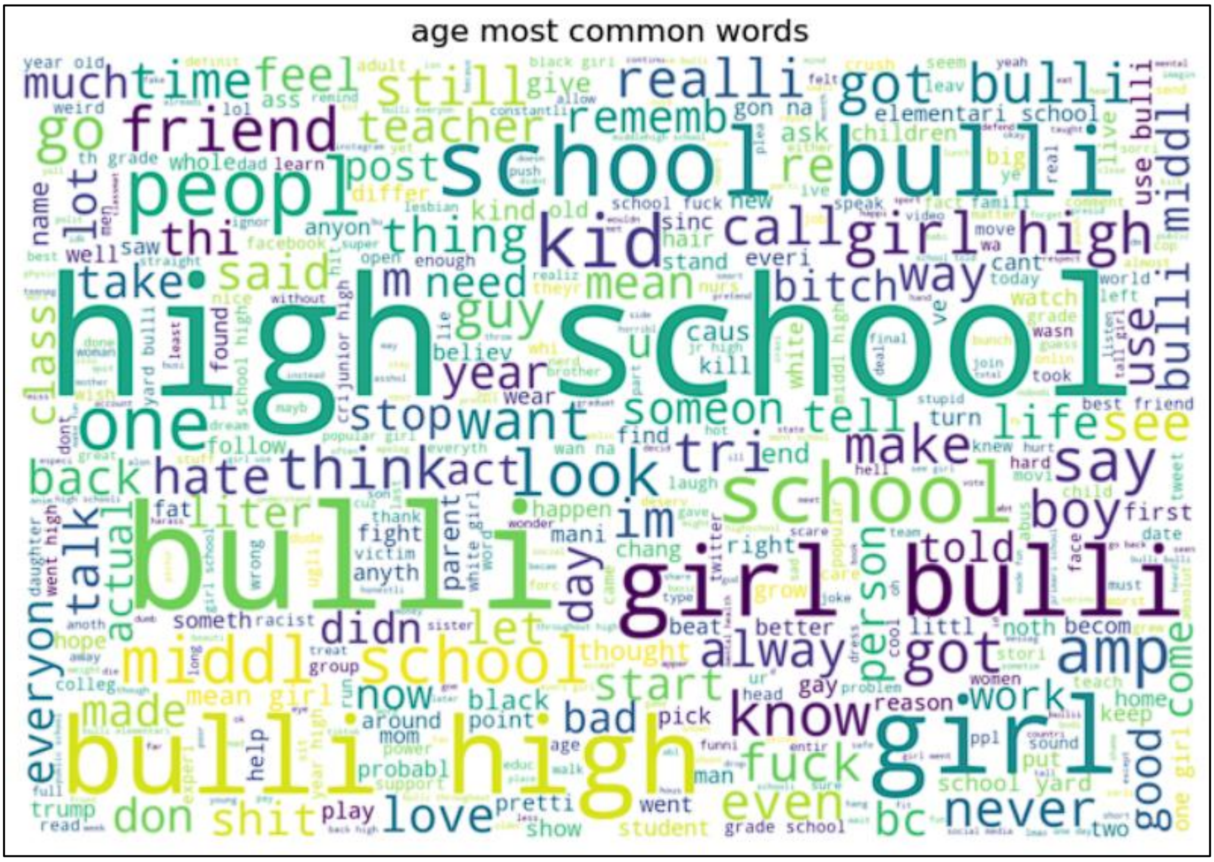
Word Clouds:

Not Cyberbullying: Illustrates the most popular words on non-cyberbullying tweets. This sets a baseline on understanding non-malicious language.

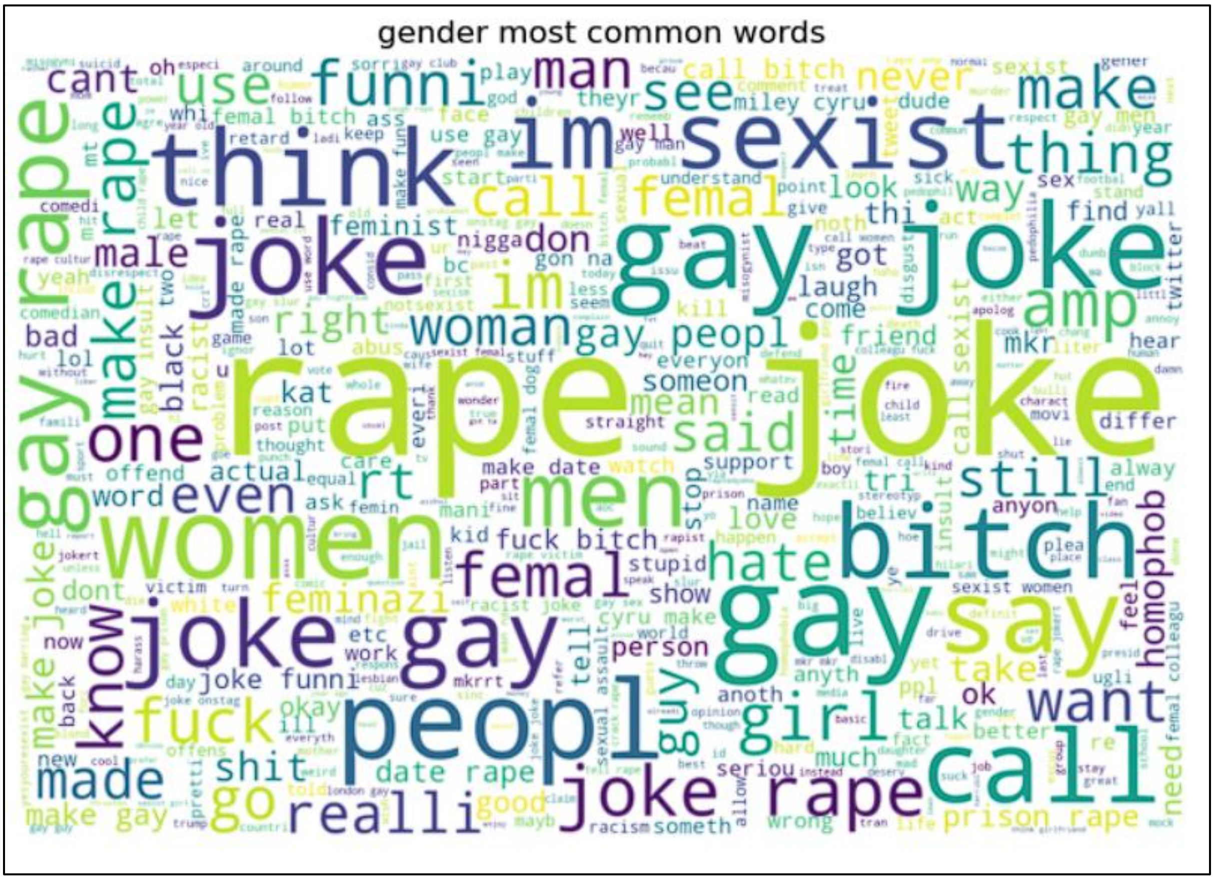
Ethnicity, Age, Religion, Gender, Other Cyberbullying: Depicts popular words that describe each kind of cyberbullying. This visualization will show the words with patterns and keywords showing abuses or intent to harm. In the "Religion" word cloud, there will be words that might illustrate themes or slurs connected with religious harassment.

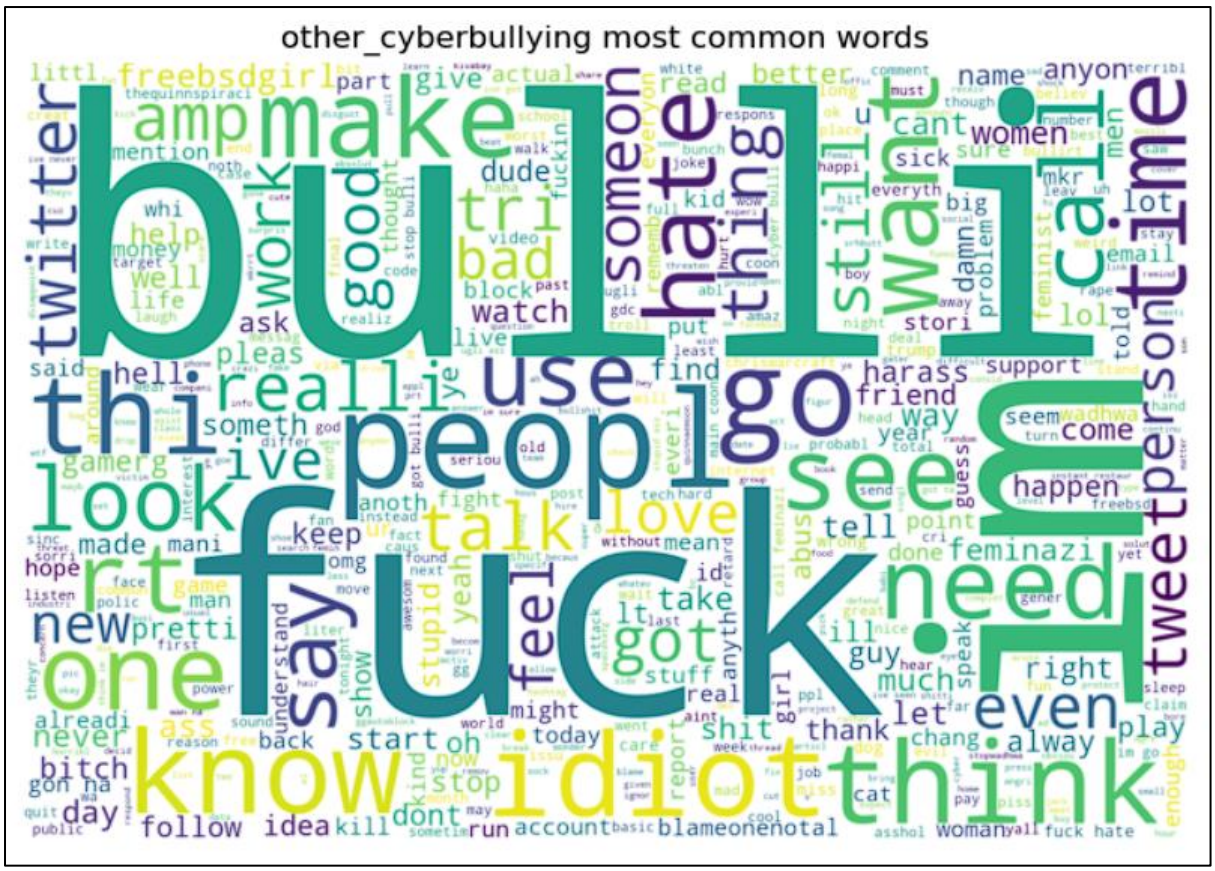



ethnicity most common words
fuck obama
nigger
obama dumb
dumb ass
ass nigger
dumb fuck
black peopl
racism
white
dumb nigger
black
racist
disgust
tayyoung fuck
fuck dumb
shit
bitch
anyth
first
say
call
said
think
one
past


age most common words
high school
school bulli
bulli
girl bulli
bulli high
middl school
peopl
friend
kid
girl
call
high
feel
still
realli
got
teacher
rememb
thing
said
take
guy
one
want
stop
someon
tell
think
look
make
say
school
boy
know
girl
got
amp
alway
love
never
even
fuck


religion most common words
muslim
idiot
islam
peopl
support
right
christian
terrorist
woman
religion
countri
hindu
think
islam
terror
know
human
kill
say
amp
india
one
want
world
terror
radic islam
hate
god
jihad
need
see
believ
trump
radic
call
good
never
christian
quran
pakistan
american
love
make
even
go
use
everi
come

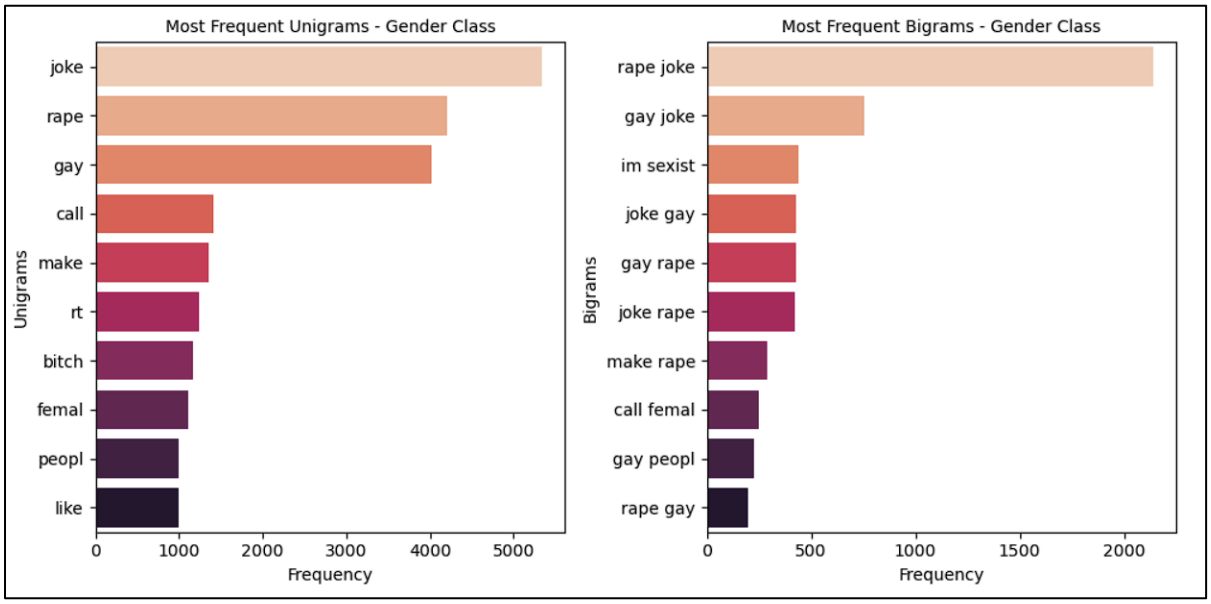

gender most common words
rape
joke
gay
sexist
think
im
joke
gay
women
bitch
peopl
call
joke gay
joke rape
men
femal
rape
make
funni
use
see
thing
call
femal
girl
say
want
know
fuck
made
go
realli
woman
one
even
hate
feminazi
still
homophob
amp
man
cant
never
make
im
right
gay
peopl
feminist
male
black
okay
shit
prison rape



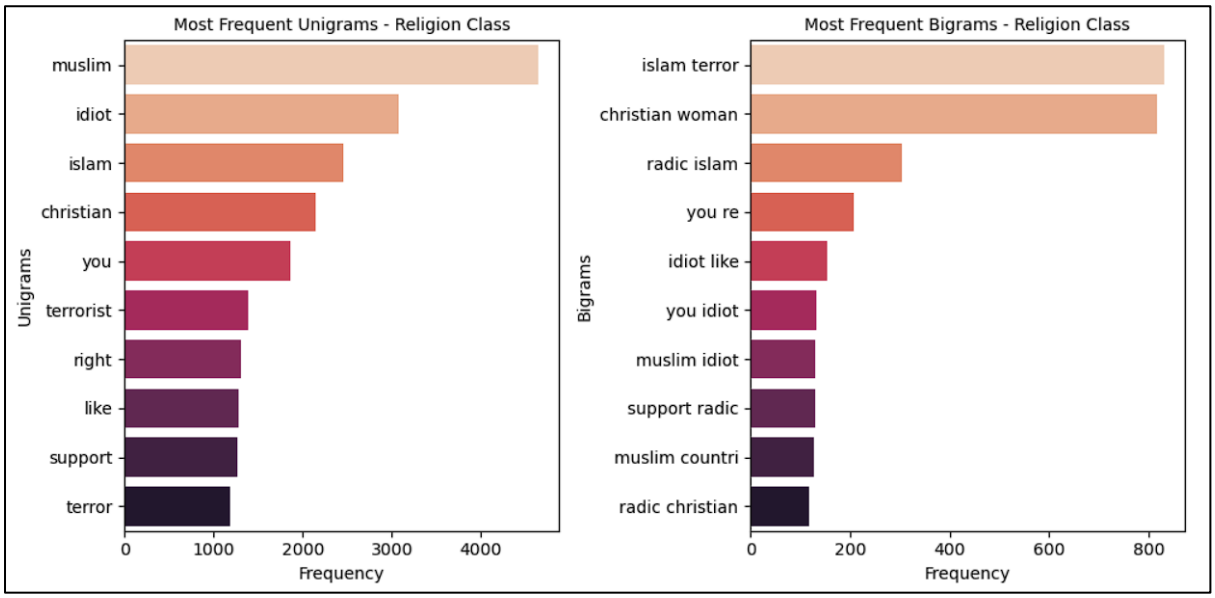
Unigrams and Bigrams:

Gender, Religion, Age, Ethnicity Classes: Analyzes individual words (unigrams) and word pairs (bigrams) specific to each cyberbullying class. These visualizations help in understanding the linguistic structure and common phrases used in different types of cyberbullying.

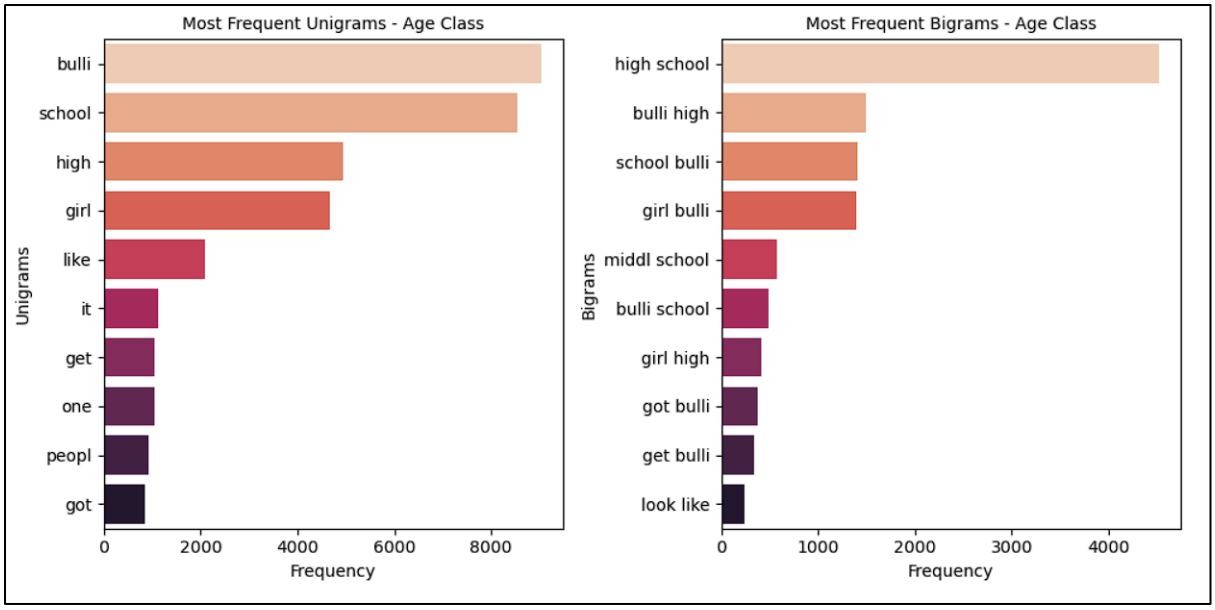
Gender Class:



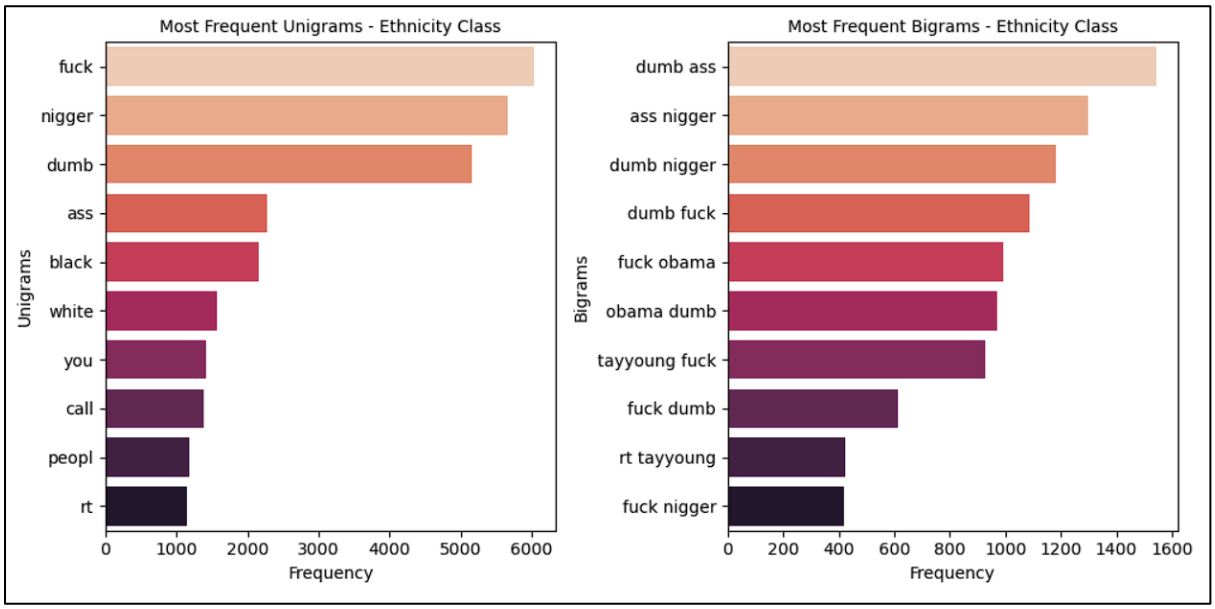
Religion Class:



Age Class:



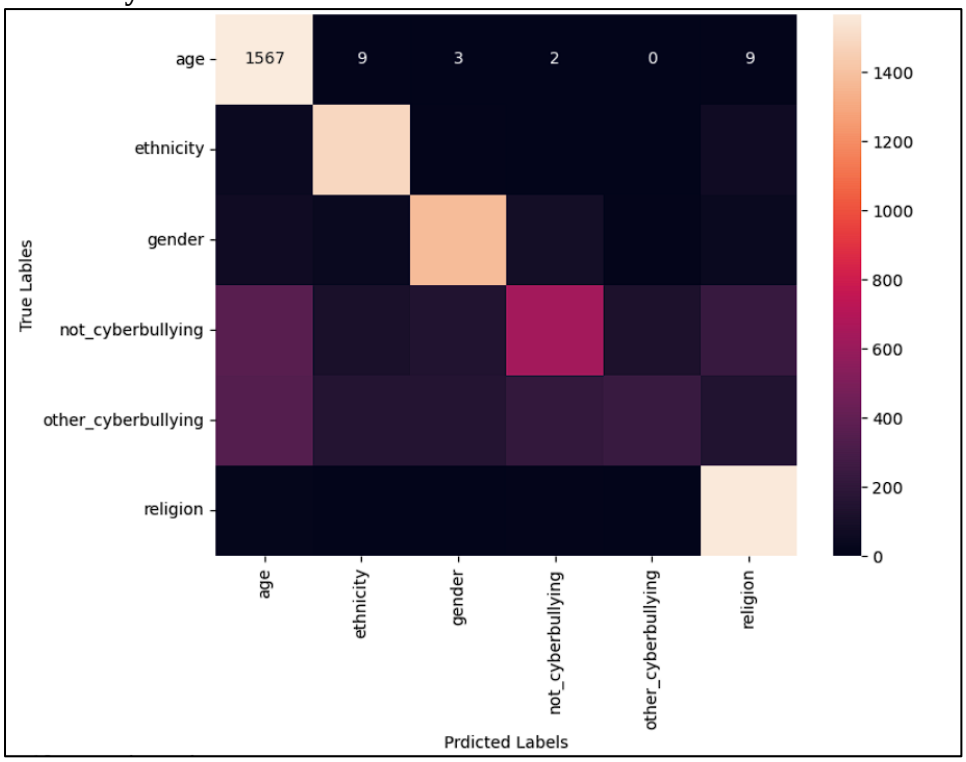
Ethnicity Class:



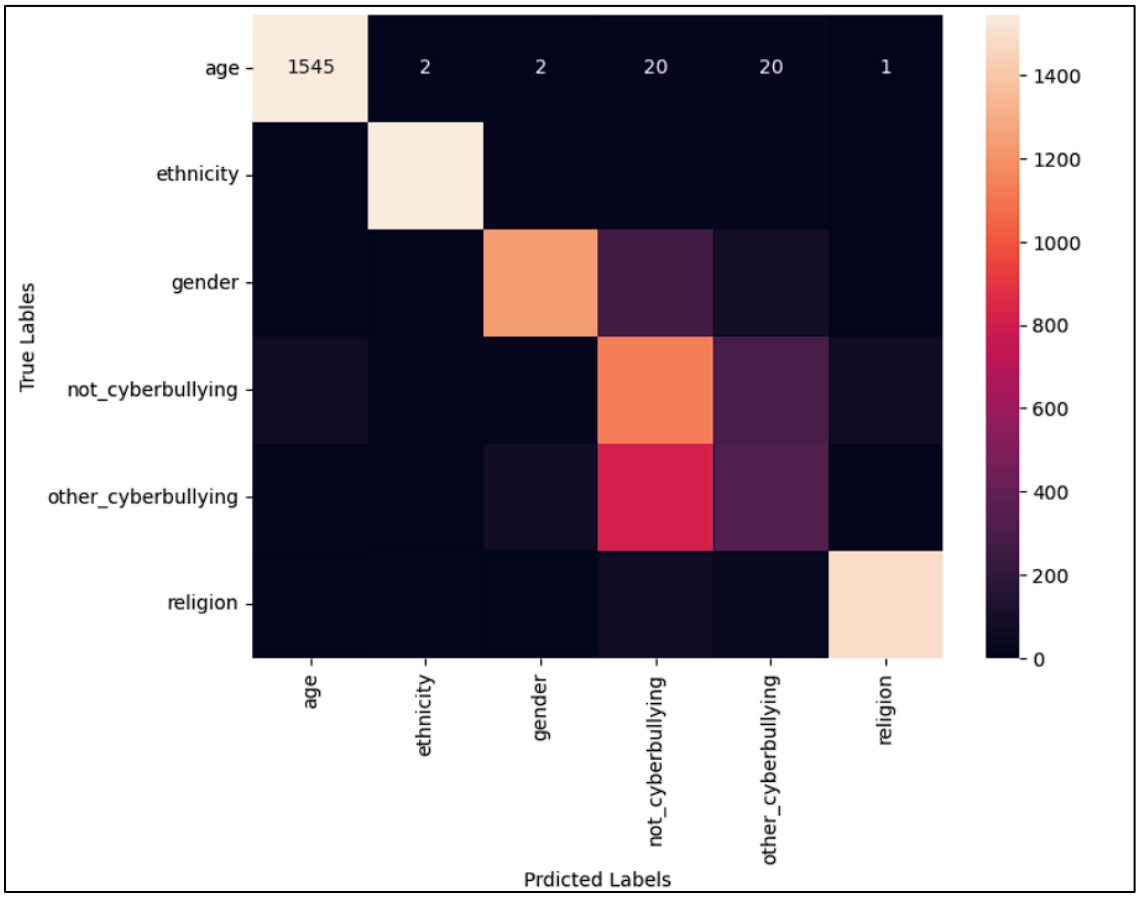
Heatmap with Confusion Matrix:

The confusion matrix for each model is represented as a heatmap showing true positive, true negative, false positive, and false negative counts. This indicates model performance in distinguishing between classes. Values on the diagonal are large, indicating that the models correctly classify the tweets with a high degree of strength. Off-diagonal values show misclassifications, and areas for improvement.

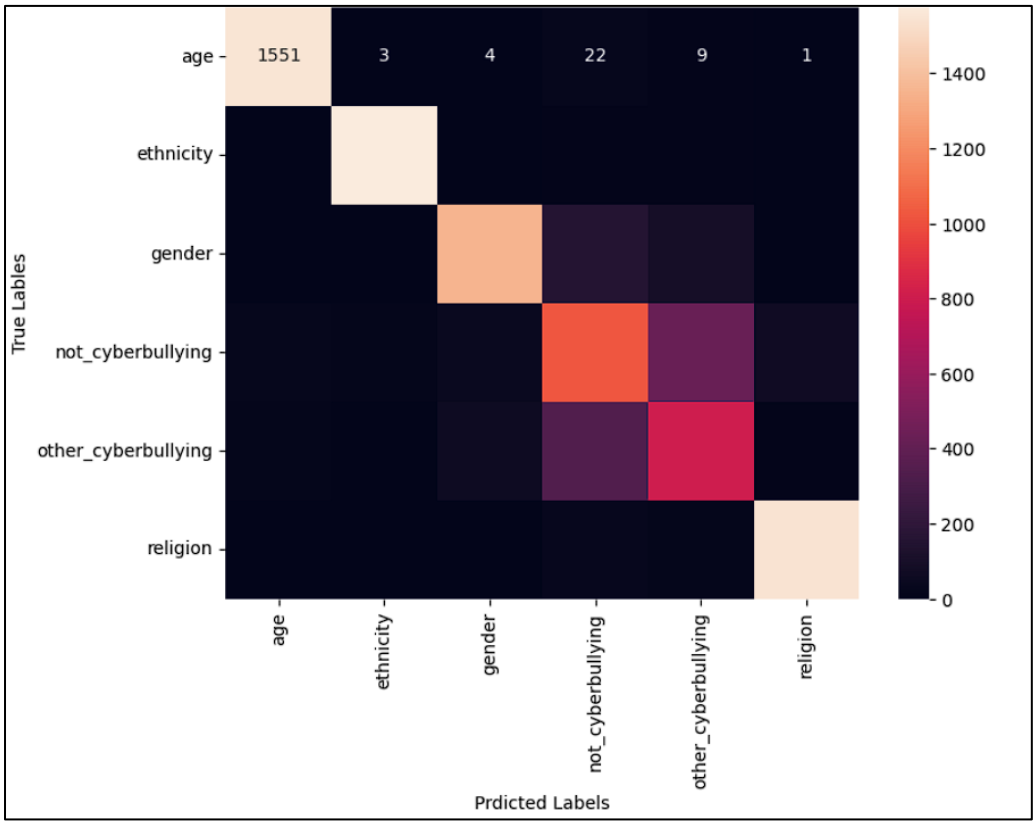
Navie Bayes:



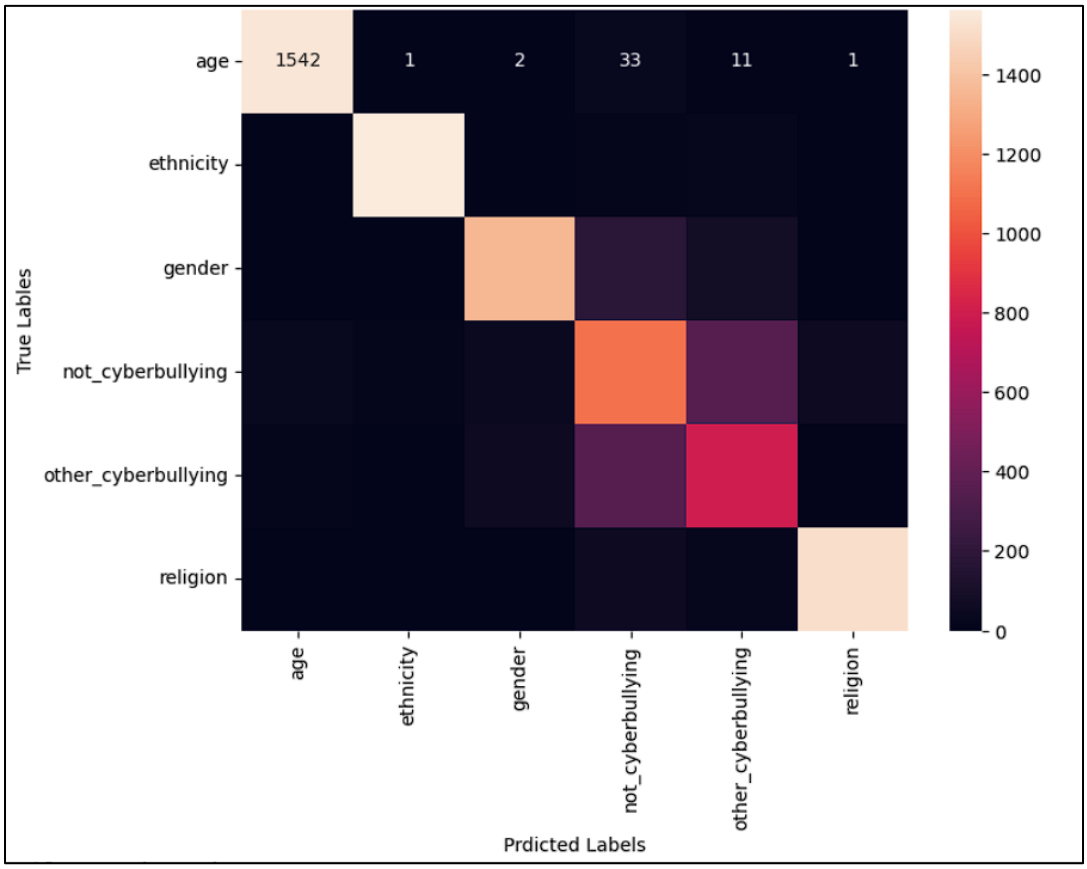
AdaBoost Classifier:



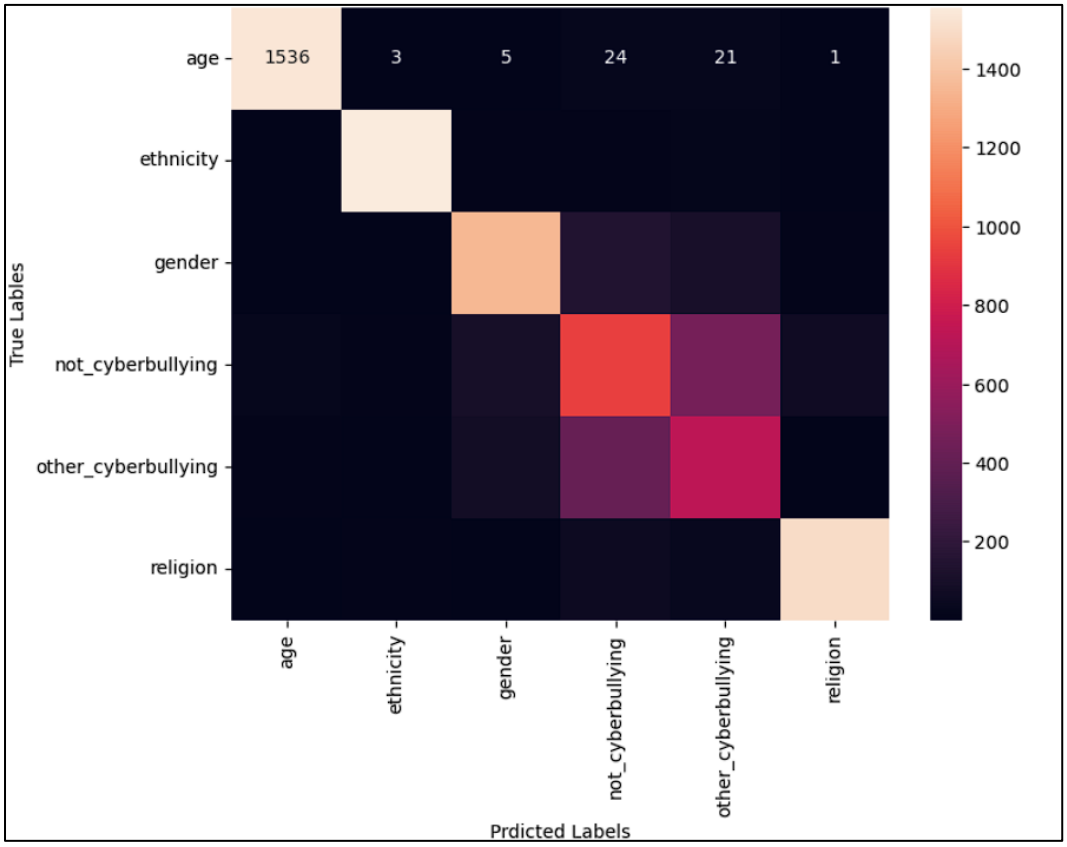
Random Forest Classifier:



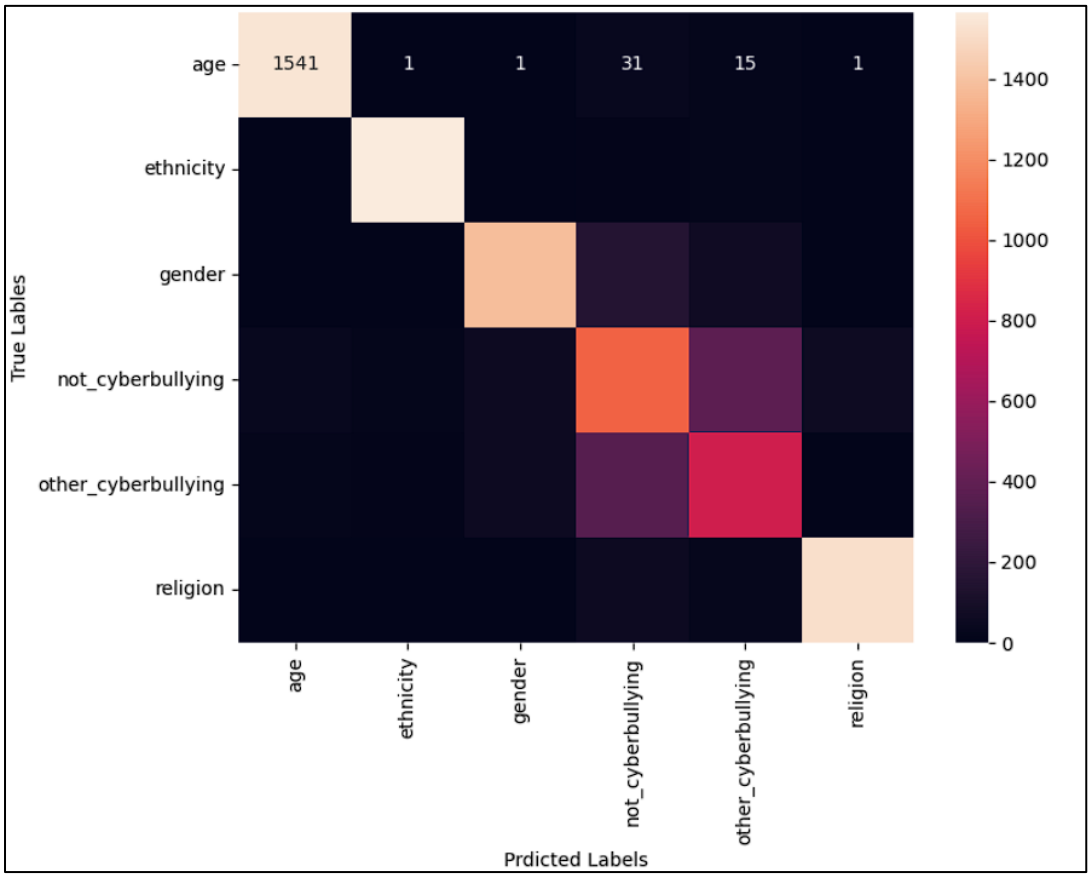
Logistic Regression:



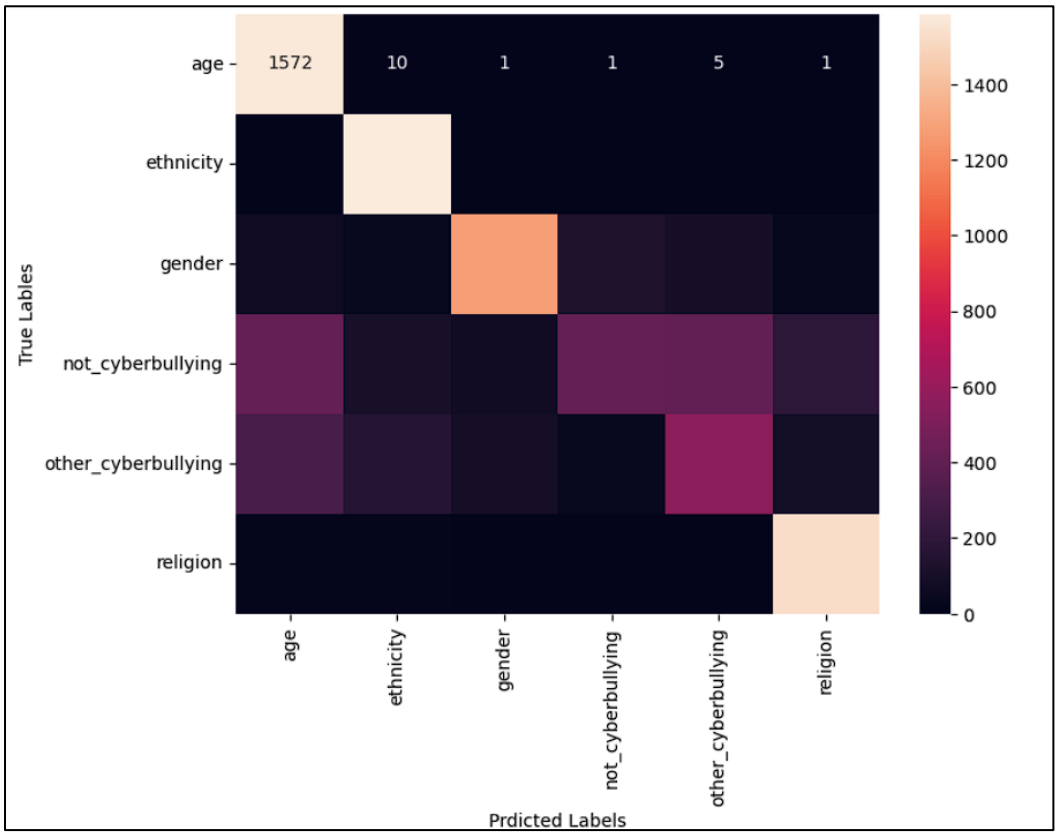
Decision Tree Classifier:



Support Vector Classifier:



K – Nearest Neighbors:



Insights from Visualizations:

- The class distribution outputs help assess the dataset's balance, informing data augmentation or re-sampling needs.
- Word clouds and n-grams provide textual insights, which are vital for feature engineering and model interpretation.
- Heatmaps summarize the strengths and weaknesses of each classifier, aiding in selecting the most robust model for deployment.
- These visualizations collectively enhance understanding of the data and model behavior, contributing to a more accurate and interpretable cyberbullying detection system.

4.2 Results Analysis:

This subsection investigates the performance of the proposed ensemble model. indicators such as confusion matrix, accuracy, precision, recall, F1-score.

Navie bayes Classifier:

Classification Report for Random Forest:

	precision	recall	f1-score	support
0	0.65	0.99	0.79	1590
1	0.83	0.93	0.87	1600
2	0.82	0.85	0.84	1605
3	0.68	0.40	0.50	1588
4	0.64	0.19	0.29	1231
5	0.76	0.98	0.86	1590
accuracy			0.74	9204
macro avg	0.73	0.72	0.69	9204
weighted avg	0.73	0.74	0.71	9204

AdaBoost Classifier:

Classification Report for Random Forest:

	precision	recall	f1-score	support
0	0.65	0.99	0.79	1590
1	0.83	0.93	0.87	1600
2	0.82	0.85	0.84	1605
3	0.68	0.40	0.50	1588
4	0.64	0.19	0.29	1231
5	0.76	0.98	0.86	1590
accuracy			0.74	9204
macro avg	0.73	0.72	0.69	9204
weighted avg	0.73	0.74	0.71	9204

Random Forest Classifier:

Classification Report for Random Forest:

	precision	recall	f1-score	support
0	0.94	0.97	0.96	1590
1	0.98	0.97	0.97	1600
2	0.92	0.77	0.84	1605
3	0.49	0.71	0.58	1588
4	0.42	0.26	0.32	1231
5	0.95	0.94	0.94	1590
accuracy			0.79	9204
macro avg	0.78	0.77	0.77	9204
weighted avg	0.80	0.79	0.79	9204

Logistic Regression:

Classification Report for Random Forest:

	precision	recall	f1-score	support
0	0.97	0.98	0.97	1590
1	0.98	0.99	0.98	1600
2	0.92	0.84	0.88	1605
3	0.65	0.64	0.65	1588
4	0.59	0.65	0.62	1231
5	0.95	0.97	0.96	1590
accuracy			0.85	9204
macro avg	0.85	0.84	0.84	9204
weighted avg	0.86	0.85	0.85	9204

Decision Tree Classifier:

Classification Report for Random Forest:

	precision	recall	f1-score	support
0	0.97	0.97	0.97	1590
1	0.98	0.98	0.98	1600
2	0.93	0.84	0.88	1605
3	0.64	0.69	0.66	1588
4	0.63	0.65	0.64	1231
5	0.96	0.95	0.95	1590
accuracy			0.85	9204
macro avg	0.85	0.85	0.85	9204
weighted avg	0.86	0.85	0.86	9204

Support Vector Classifier:

```
Classification Report for Random Forest:
              precision    recall  f1-score   support

           0       0.96      0.97      0.97      1590
           1       0.98      0.98      0.98      1600
           2       0.93      0.86      0.89      1605
           3       0.64      0.66      0.65      1588
           4       0.62      0.65      0.63      1231
           5       0.96      0.95      0.95      1590

    accuracy                           0.85      9204
   macro avg       0.85      0.85      0.85      9204
weighted avg       0.86      0.85      0.85      9204
```

K- Nearest Neighbors:

```
Classification Report for Random Forest:
              precision    recall  f1-score   support

           0       0.66      0.99      0.79      1590
           1       0.83      0.99      0.90      1600
           2       0.88      0.79      0.83      1605
           3       0.72      0.26      0.38      1588
           4       0.52      0.45      0.49      1231
           5       0.83      0.96      0.89      1590

    accuracy                           0.75      9204
   macro avg       0.74      0.74      0.72      9204
weighted avg       0.75      0.75      0.72      9204
```

Accuracy Obtained:

	Model	Accuracy
0	Navie Bayes Classifier	0.743807
1	AdaBoost Classifer	0.789222
2	Random Forest Classifier	0.852130
3	Logistic Regression	0.854085
4	Decision Tree Classifier	0.824533
5	Support Vector Machine	0.853216
6	K-Nearest Neigbors	0.753151

CONCLUSION

This developed project on a cyberbullying tweet classifier demonstrates the full potential of machine learning and Natural Language Processing in dealing with negative online behavior. By incorporating data preprocessing, feature extraction, and model implementation within the methodology used, this project efficiently detects tweets with cyberbullying content. Different machine learning models were tested, and the ones that worked well were Random Forest Classifier, Logistic Regression, and Support Vector Machine with an accuracy of 85%, which outperformed other models such as Naive Bayes, AdaBoost, Decision Tree Classifier, and K-Nearest Neighbors. This indicates that strong preprocessing techniques like removal of stopwords, tokenization, and vectorization through Count Vectorizer and TF-IDF Vectorizer are required to improve the performance of models. Exploratory data analysis and visualization, such as word clouds, n-grams, and confusion matrices, helped immensely in understanding the dataset and behavior of models. Although the achieved accuracy is good, there is still room for improvement. Future work can include exploration of deeper learning models, such as LSTMs or transformers, and incorporation of more features or even fine-tuning of hyperparameters. However, this project is an important step toward making the online space safer by automatically detecting cyberbullying content, thus encouraging a more inclusive and respectful digital environment.

REFERENCES

[1] Cyberbullying Detection and Machine Learning: A Systematic Literature Review. This article examines scholarly publications from 2011 to 2022 on cyberbullying detection using machine learning, providing insights into various approaches and their effectiveness. https://link.springer.com/article/10.1007/s10462-023-10553-w?utm_source=

[2] Natural Language Processing for Cyberbullying Detection. This paper discusses the application of NLP techniques in identifying offensive content, highlighting methods for processing and analyzing textual data in the context of cyberbullying. https://www.ijcjournal.org/index.php/InternationalJournalOfComputer/article/view/2136?utm_source=

[3] This project has been done under Jupyter Notebook Interpreter. https://jupyter.org/install

[4] Natural Language Processing has been installed to work on cyberbullying tweet classifier. https://medium.com/pythoneers/basics-of-natural-language-processing-in-10-minutes-2ed51e6d5d32#:~:text=To%20install%20Jupyter%20notebook%2C%20just,are%20going%20to%20learn%20about