Detecting Hate Speech on Twitter

Xinyu Wang

July 30, 2023

Contents

1 Background

In an age where social media has become a powerful tool for communication and information dissemination, it's vital to understand the nature of the content shared on these platforms. Among these, Twitter stands out due to its significant user base and the brevity of its content, leading to unique patterns of usage and interaction. However, alongside the positives, negative aspects such as the spread of hate speech have become an increasing concern.

Hate speech, defined as any form of communication that discriminates, belittles, or harms an individual or a group based on attributes such as race, religion, ethnic origin, sexual orientation, disability, or gender, has seen a significant rise on social platforms like Twitter. It poses a severe threat to online safety and community well-being, warranting thorough investigation and efficient handling.

This project aims to develop and test machine learning models that can accurately identify hate speech in tweets, thereby providing a robust tool to combat this issue. In turn, these models could help in maintaining a healthier, more positive environment on Twitter. The data used for the project comprises a vast number of tweets, some of which are marked as hate speech, allowing for a comprehensive analysis and evaluation of the developed models.

For the sake of clarity, the report is divided into sections discussing data collection, exploratory data analysis, model development (including Logistic Regression, Random Forests, Neural Network and BERT models), and evaluation of these models.

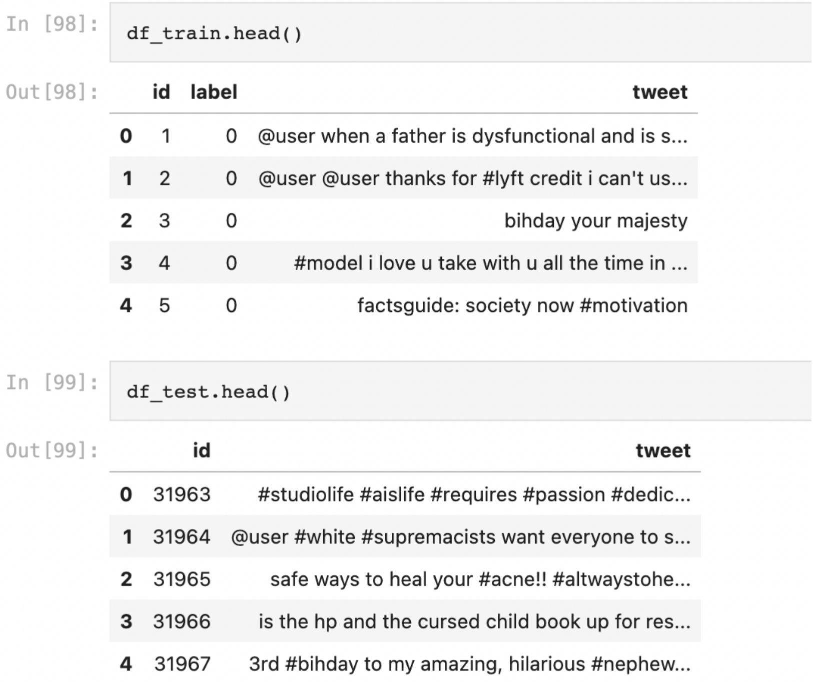
2 Data Collection and Exploration

The project began with a comprehensive data collection phase from Twitter. A total of 49790 tweets were collated, consisting of both a training set where each tweet was labeled as either containing hate speech or not, and a test set where our developed models would be applied. A notable observation was the imbalance in the data, with a relatively small fraction of tweets being classified as hate speech. This suggests an overall positive environment on Twitter, implying that only a minority of tweets require intervention.

Moreover, an interesting trend was that the majority of the tweets identified as hate speech contained politically-charged words such as ’Trump’, ’Obama’, and terms related to social inequalities like ’white’, ’black’, ’women’. This pattern underscores the role of socio-political discourse in the propagation of hate speech.

Twitter, by its very nature, encourages robust interaction among users. As evident from the data, some users had higher frequencies of hate speech usage than others. Detailed information about these trends can be found in the Exploratory Data Analysis (EDA) section of this report.

To provide a tangible understanding of the dataset structure, an extract of the data is displayed below:



```
In [98]:  df_train.head()

Out[98]:
      id  label                                            tweet
   0   1      0   @user when a father is dysfunctional and is s...
   1   2      0  @user @user thanks for #lyft credit i can't us...
   2   3      0                                bihday your majesty
   3   4      0     #model i love u take with u all the time in ...
   4   5      0                  factsguide: society now #motivation

In [99]:  df_test.head()

Out[99]:
         id                                              tweet
   0  31963   #studiolife #aislife #requires #passion #dedic...
   1  31964  @user #white #supremacists want everyone to s...
   2  31965       safe ways to heal your #acne!! #altwaystohe...
   3  31966    is the hp and the cursed child book up for res...
   4  31967  3rd #bihday to my amazing, hilarious #nephew...
```

Figure 1: Snapshot of the Training Data

This snapshot of the training data showcases the features extracted from the tweets and the corresponding labels assigned, forming the basis for our predictive models.

3 Data Preprocessing

3.1 Data Cleaning

The preprocessing phase involved multiple steps to ensure the tweets were properly cleaned and formatted for use in the machine learning models.

1. **Lowercasing**: The first step was standardizing the case by converting all the text in the tweets to lowercase. This helps ensure uniformity and prevents the model from treating the same words with different cases as distinct.

2. **Remove Noise**: The next step was to clean the tweets by removing any noise. This includes numbers, punctuation, and special characters. Such elements do not carry significant meaning in the context of identifying hate speech and their removal helps to reduce the complexity of the dataset.

3. **Tokenization**: This process involves breaking down the tweet into individual words or tokens. The Natural Language Toolkit (NLTK) library was used for this purpose. Tokenization transforms the text into a form that can be easily analyzed and is a crucial step in Natural Language Processing (NLP).

4. **Stopword Removal**: This step involves removing common, non-informative words that do not contribute to the sentiment of the tweet. Stopwords are words that, while necessary for human communication, do not provide significant meaning to machine learning models and thus can be removed.

5. **Stemming**: Stemming is the process of reducing words to their root or base form. For example, ”running”, ”runs”, and ”ran” would all be reduced to ”run”. This helps to simplify the data and reduce the number of unique words the model needs to consider. The NLTK library provides methods for this step.

6. **Rejoining**: The final step involves rejoining the processed words into a single string. This recombined text is then used as the input for the machine learning models.

Upon completion of these preprocessing steps for, the data was in a suitable format for training the machine learning models. The cleaned, tokenized, and lemmatized tweets now formed a dataset that could accurately represent the nature of hate speech on Twitter.

3.2 Vectorization

To transform the preprocessed textual data into a format that could be utilized by machine learning algorithms, a process of vectorization was performed. Various techniques were adopted to transform the tweets into meaningful numeric vectors representing each document.

1. **Term Frequency-Inverse Document Frequency (TF-IDF)**: The initial method utilized for vectorization was TF-IDF. It provides a way of scoring the importance of words (or terms) in the document based on how frequently they appear across multiple documents. The TF-IDF approach was set to retain the top 500 features (words) in the vectorized output.

2. **Word2Vec**: The second method of vectorization employed was Word2Vec. This technique generates vector representations of words in a way that preserves their semantic relationships. Again, the configuration was set to maintain the top 500 features.

3. **BERT (Bidirectional Encoder Representations from Transformers)**: Finally, the BERT model was employed for vectorization. BERT, a transformer-based machine learning technique for natural language processing, was used to generate high-dimensional word embeddings. The BERT model produced vectors of length 768 for each tweet.

These various vectorization methods yielded sets of features which were then utilized in training various machine learning models, namely Logistic Regression, Random Forest, and Neural Networks, as detailed in the subsequent section on Model Training.

4 Exploratory Data Analysis

The Exploratory Data Analysis (EDA) stage involved a comprehensive investigation of the dataset. The primary aim was to understand the data, identify patterns, detect anomalies, test hypotheses, and verify assumptions.

4.1 Dataset Distribution

We first looked at the distribution of the dataset. Since the data was imbalanced, with a relatively small fraction(7.014%) of tweets being classified as hate speech, this posed an interesting challenge for our machine learning models. The distribution is shown in the pie chart below:

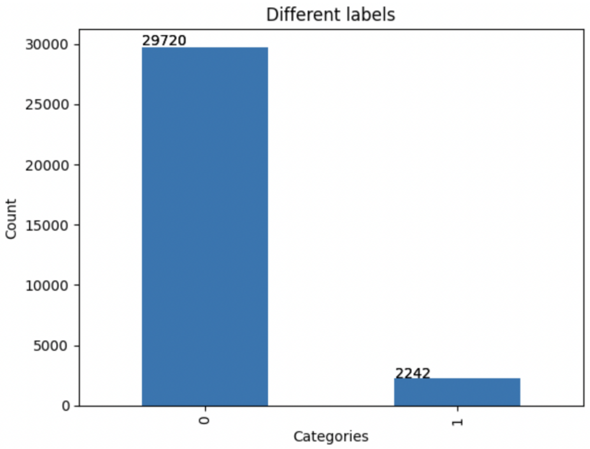
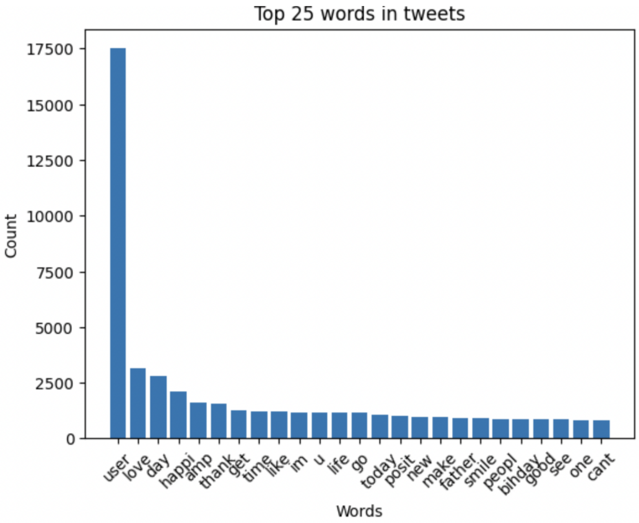


Figure 2: Dataset Distribution

4.2 Most Frequent Words

n our study, to discern and visualize the most frequent words, we constructed a bar plot representing the top 25 most recurring words, along with the top 10 most prevalent bigrams. As evident from Figure 3, the list of top 25 words includes ’user’, ’love’, ’day’, ’happy’, ’amp’, ’thank’, ’get’, ’time’, and ’like’. These keywords suggest the inherently interactive nature of Twitter and convey the general atmosphere of positivity and amicability that pervades the platform.

When it comes to the bar plot of the top 10 most frequent bigrams, the dominant pairs include 'user_user', 'love_day', etc., which further underscore the characteristics of the Twitter environment.



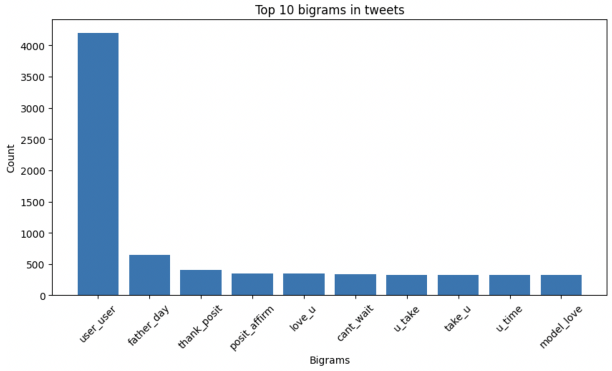


Figure 3: Word Clouds for Hate Speech (left) and Non-Hate Speech (right)

A word cloud was then generated to visualize the most frequent words in the tweets. This was performed separately for both hate speech and non-hate speech categories.

The findings from the word clouds echo the patterns previously observed in the bar plots. After removing words like 'user' and 'amp'—common stop-words in the nltk library—the word cloud of the entire tweet corpus highlights a significant frequency of positive words such as 'love', 'happy', 'day', 'new', and 'thank'. This further underlines the general mood of positivity that

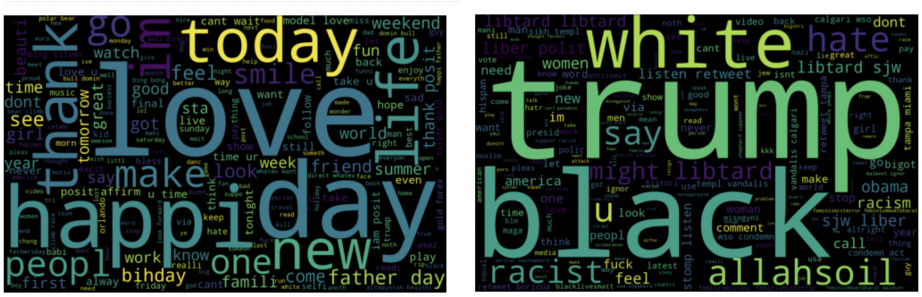


Figure 4: Word Clouds for Hate Speech (left) and Non-Hate Speech (right)

characterizes the majority of Twitter interactions.

However, the word cloud derived from hate speech tweets paints a different picture. It is dominated by politically charged words that appear most frequently in the training dataset. Moreover, terms like 'white', 'black', and 'racist' stand out, suggesting the presence of racial disparities, which starkly contrast with the values of equality embraced by the United States. Disturbingly, there are words such as 'allahsoil' and 'libertard' that are inherently offensive and constitute hate speech.

To maintain the friendly and positive environment that Twitter strives for, it's crucial to identify and remove such instances of hate speech. By doing so, we contribute to fostering a safer and more inclusive digital space.

4.3 Topic Explore

In our analysis, we first preprocessed the tweet data, creating a Document-Term Matrix while excluding common English stop words, including terms such as 'user' and 'amp'. This was the foundational step to structure our text data effectively for the subsequent analysis.

We then deployed the Latent Dirichlet Allocation (LDA) algorithm, segmenting the entire tweet corpus into ten distinct topics. To gain deeper insights into the nature of each topic, we examined the top ten words ranked by frequency under each topic.

The results revealed that each topic possesses its unique lexical attributes. For example, Topic #0 primarily features words like 'father', 'birthday', 'love', and 'happy', potentially mirroring Twitter discussions about family and holiday celebrations. In contrast, Topic #2 is characterized by terms such as 'sex', 'black', 'race', and 'bear', suggesting conversations related to social issues or the natural environment. Furthermore, Topic #6, with prominent words like 'beach', 'sun', 'Trump', 'Orlando', and 'summer', appears to be reflective of dialogues on travel, politics, and seasonal transitions.

To supplement these textual insights, we created a visual representations—charts highlighting the key words for each topic, which aligns with the results(Figure 5).

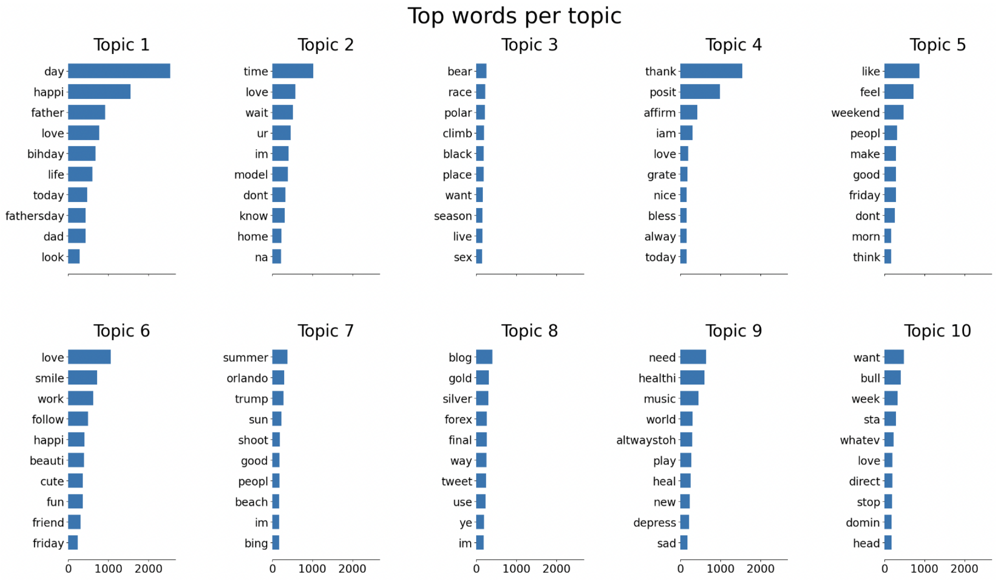


Figure 5: Top words of each topic of all tweets

Following a similar procedure as before, we delved into the sub-section of our data specifically targeting hate speech. We preprocessed the tweets labeled as hate speech, excluding the same set of English stop words, and applied the LDA algorithm to this subset of data, generating ten topics.

Upon examination of the ten most frequent words under each hate speech topic, we observed unique lexical patterns indicative of the content of such speech. For example, Topic #0 features terms like 'racism', 'race', 'black', 'white', likely corresponding to racially-charged discourse. Topic #2, highlighted by words such as 'equal', 'right', 'sjw', 'libtard', appears to involve politically-charged conversations. Topic #5's most frequent words, including 'maga', 'misogyny', 'bigotry', 'trump', seem to denote discussions with explicit political overtones or those dealing with gender biases. We also did the visualization for this part(Figure 6).

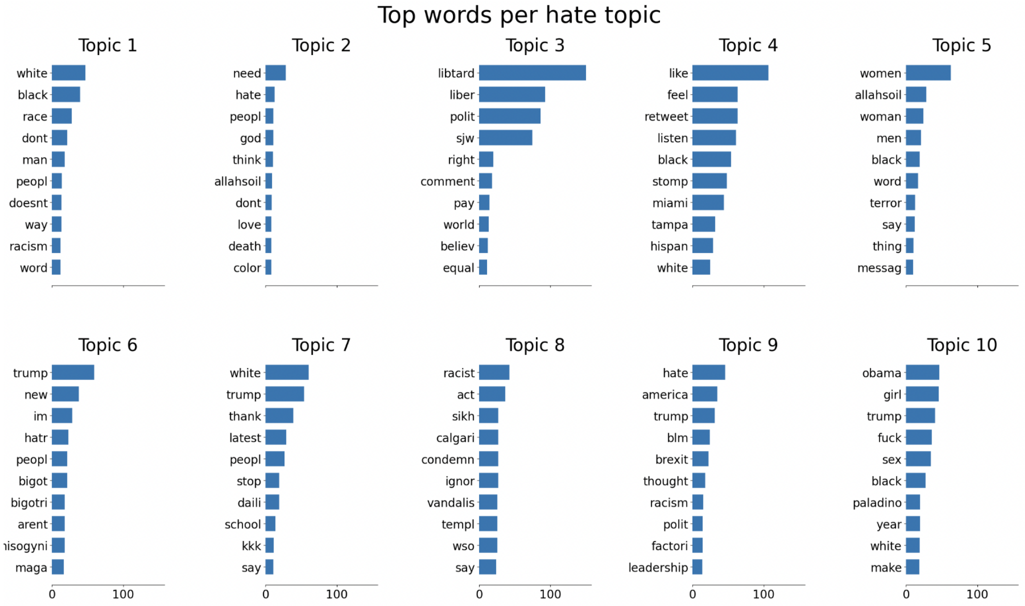


Figure 6: Top words of each topic of hate speeches

5 Model Training: Logistic Regression and Random Forests

5.1 Logistic Regression

We trained a Logistic Regression model on our data after scaling the features to have zero mean and unit variance. The model was implemented with a Stochastic Gradient Descent (SGD) classifier using the log loss function, and class weights were balanced to adjust for the class imbalance in our data. The accuracy of the model was about 81%, and the ROC-AUC score was approximately 0.87, indicating a good trade-off between sensitivity and specificity.

The precision, recall, and F1-score for the negative class (non-hate speech) were fairly high. However, for the positive class (hate speech), the precision was low, but the recall was reasonably good. This indicates that the model was able to capture most of the hate speech tweets, but at the same time, it also misclassified some non-hate speech tweets as hate speech.

5.2 Random Forest

We also implemented a Random Forest classifier, but before that, we used Synthetic Minority Over-sampling Technique (SMOTE) to tackle the issue of class imbalance by creating synthetic samples of the minority class. We then trained the Random Forest model on the oversampled data. The accuracy of this model was higher (around 95

The precision, recall, and F1-score for the negative class were high, indicating good performance. For the positive class, the precision was considerably better than the Logistic Regression model, but the recall was lower. This means that the model was more precise in identifying hate speech tweets, but it missed more actual hate speech tweets compared to the Logistic Regression model.

In both models, we plotted the ROC curve, which demonstrates the performance of a binary classifier as its discrimination threshold is varied.

6 Model Training: Neural Network and BERT

6.1 Neural Network

We also implemented a neural network model, but before that, we conducted some preprocessing steps. Firstly, we transformed the tweets into numerical matrices using the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm. Then, we applied Principal Component Analysis (PCA) to reduce the dimensionality of the features. Our neural network model consisted of two hidden layers and utilized ReLU as the activation function.

The accuracy of this neural network model was high (around 94%), and the ROC-AUC score was also higher, approximately 0.92. The precision, recall, and F1-score for the negative class were high, indicating excellent performance in identifying non-hate speech tweets. For the positive class, although the precision was better than both the Logistic Regression and Random Forest models, the recall was lower, suggesting that the model had room for improvement in identifying actual hate speech tweets.

Overall, although the neural network model had high accuracy and ROC-AUC scores, it had a lower recall rate for hate speech tweets. This suggests that our focus in further optimizing the model should be on improving the recall for the positive class. We could attempt to enhance the recall rate by adjusting the parameters of the neural network, such as increasing the number of hidden layers or changing the activation function. At the same time, we can also consider using different feature extraction and dimensionality reduction methods to further improve the performance of the model.

6.2 BERT

For the next experiment, we implemented a model utilizing the BERT (Bidirectional Encoder Representations from Transformers) architecture. Due to the string data of tweets, this model was suitable as BERT is designed to understand the semantics of sentences directly, without the need for manually engineered features. We ran the model on a T4 GPU, which allowed for efficient model training.

In dealing with the imbalanced data problem, we defined and implemented class weights. Specifically, we increased the weight of the minority class (hate speech) by a factor of 100 in the loss calculation. This way, the model would pay more attention to correctly classifying these instances during training. The learning parameters were set to 5 epochs, a batch size of 32, learning rate of 3e-5, epsilon of 1e-08, and clipnorm of 1.0.

The BERT model yielded an ROC-AUC score of 93%, marking a significant improvement from the previous models. This result indicates that the BERT model was quite effective in distinguishing between hate speech and non-hate speech tweets, suggesting its potential to be optimized further. Future efforts might be directed towards fine-tuning this model and exploring its performance with different learning parameters or training strategies.

7 Model Evaluation

To evaluate the performance of our models, we utilized both accuracy and ROC-AUC scores. Accuracy, which calculates the proportion of correct predictions out of all predictions made, provided a straightforward measure of model performance. However, given the imbalanced nature of our dataset, accuracy could be a misleading metric.

To address this, we also considered the ROC-AUC score. Unlike accuracy, the ROC-AUC score, which stands for Receiver Operating Characteristic - Area Under the Curve, takes both the true positive rate (sensitivity) and the false positive rate (1 - specificity) into account. This makes it particularly useful for evaluating model performance in the context of imbalanced datasets.

Here are the results of our models:
- The Logistic Regression model achieved an accuracy of 0.81 and an ROC-AUC score of approximately 0.87.
- The Random Forest model achieved an accuracy of 0.85 and an ROC-AUC score of approximately 0.90.
- The Neural Network model obtained an ROC-AUC score of approximately 0.91.
- The BERT model achieved an ROC-AUC score of 0.93.

The ROC-AUC scores for all the models were satisfactory, indicating good performance in distinguishing between hate speech and non-hate speech tweets. However, it's worth noting that due to the complex and evolving nature of language use on social media platforms, occasional misclassifications are inevitable.

In addition, we have included the ROC-AUC curves(Figure 7) of all models for visual comparison. This should provide a more intuitive understanding of the trade-off between sensitivity and specificity that each model achieves.

8 Conclusion

In conclusion, the Logistic Regression, Random Forest, Neural Network, and BERT models all succeeded in classifying tweets as hate speech or non-hate speech with high accuracy. This study could significantly contribute to de-

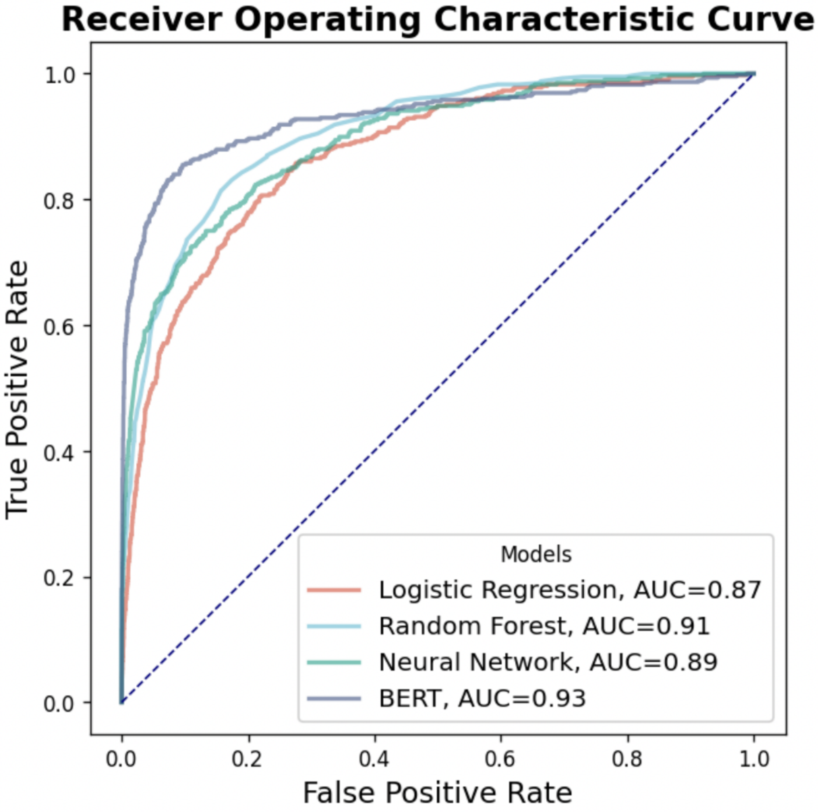


Figure 7: Top words of each topic of hate speeches

tecting and filtering out hate speech on social media platforms like Twitter, thereby creating safer and more inclusive digital spaces. Continuous monitoring and periodic updates are recommended to ensure the models' adaptation to the dynamic nature of language on social media platforms.