IST 736 : Text Mining

Final Project Report

Analysing Hate Speech Dynamics on Twitter Using Text Mining and NLP Techniques

Team Members

Maruthamuthu K Sowmeya *Gughapriyaa Elango*

Mohammed Huzaif Kherani *Harshita Umesh Tanksali*

PROJECT OVERVIEW

The project focuses on tackling the prevalence of hate speech on social media, aiming to create a robust model for detecting and flagging such harmful content. Primarily leveraging cutting-edge techniques like Convolutional Neural Networks (CNN), it aims to develop an effective hate speech detection system that contributes to a safer online environment. Beyond just user protection, the project prioritizes algorithmic fairness, striving to minimize biases in hate speech classification and ensure equitable treatment across diverse user groups. Additionally, it aims to foster community engagement by facilitating discussions on responsible AI usage, ethical considerations, and the collaborative approach necessary to combat online toxicity effectively. Overall, the project's holistic approach addresses both technical aspects of model development and the ethical implications, emphasizing the importance of safeguarding user well-being and promoting a positive and inclusive online community.

METHODOLOGY

In our methodology, we systematically approached hate speech classification in Twitter data. We initiated Data Loading and Exploration, scrutinizing dataset structure and characteristics. Text Pre-processing involved cleaning, normalizing, and tokenizing text for subsequent analysis. Leveraging Word Cloud Visualization and Hashtag Analysis, we visualized word frequency and explored hashtag prevalence and patterns. Topic Modelling, employing Latent Dirichlet Allocation (LDA), identified dominant topics, while Sentiment Analysis categorized tweets. Vader Sentiment Analysis enriched sentiment understanding. Machine Learning Models were trained and evaluated for hate speech classification, with model deployment for real-time analysis of new textual data. Ethical considerations underpinned the entire process, emphasizing responsible data handling and fair model training to mitigate biases and ensure a comprehensive and conscientious approach to the complexities of hate speech on Twitter.

Activities	**Subtasks**
Data Loading and Exploration	Load Dataset Explore Dataset Structure Examine Data Characteristics
Text Pre-processing	Clean Text Data Tokenize and Normalize Text
Word Cloud Visualization	Generate Word Clouds Analyse Word Frequency
Hashtag Analysis	Analyse Hashtag Prevalence Examine Hashtag Usage Patterns
Topic Modelling (LDA)	Apply LDA for Topic Extraction Identify Dominant Topics
Sentiment Analysis	Perform Sentiment Analysis Categorize Sentiment
Vader Sentiment Analysis	Utilize Vader for Polarity Analysis Enhance Sentiment Understanding
Machine Learning Models	Train Various ML Models Evaluate Model Performance
Deploying the Models	Implement Models for Real-time Analysis Classify New Textual Data
Ethics	Ensure Responsible Data Handling Maintain Fair Model Training

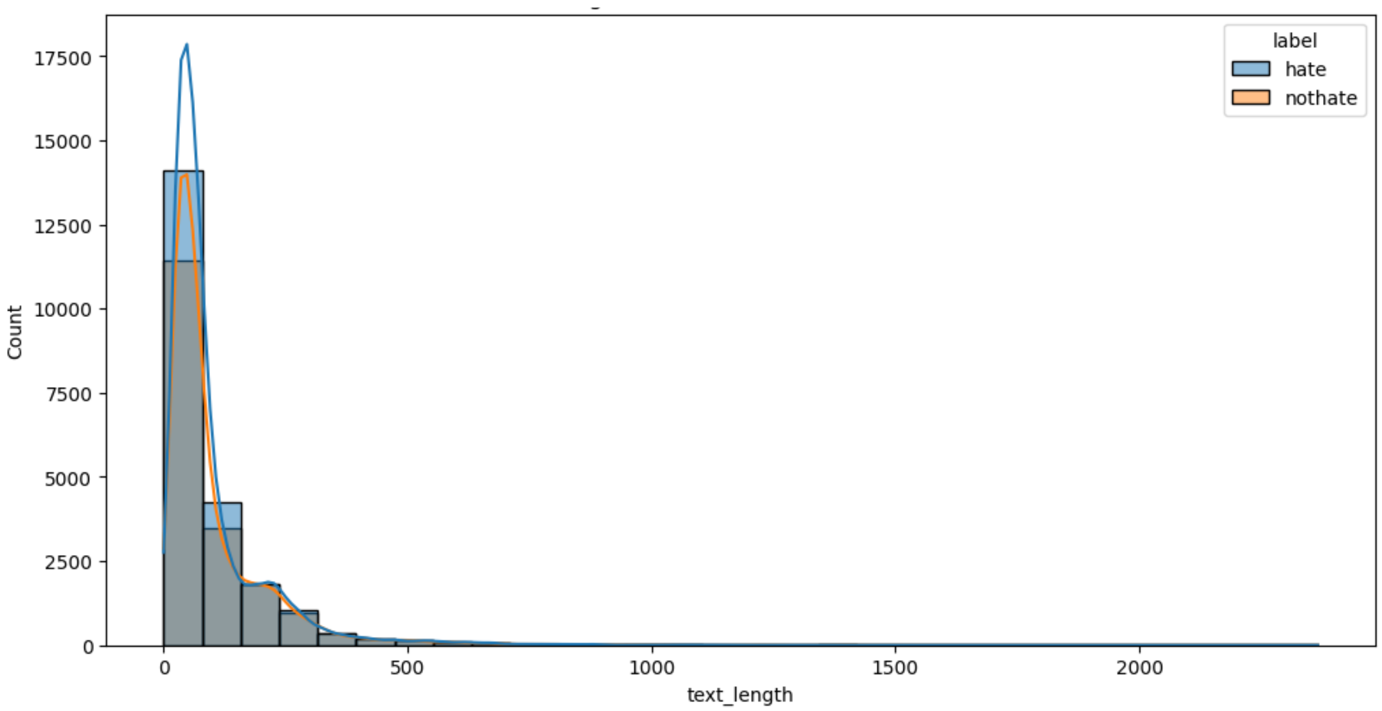
DATASET DESCRIPTION

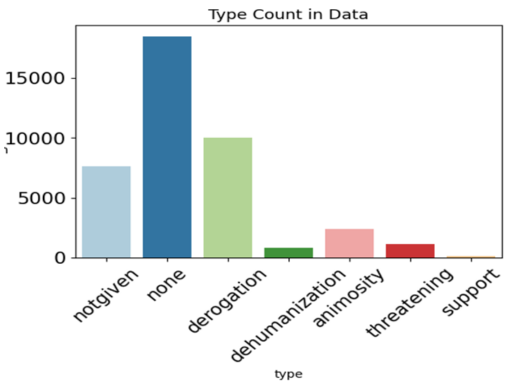
Entries Dataset Description:

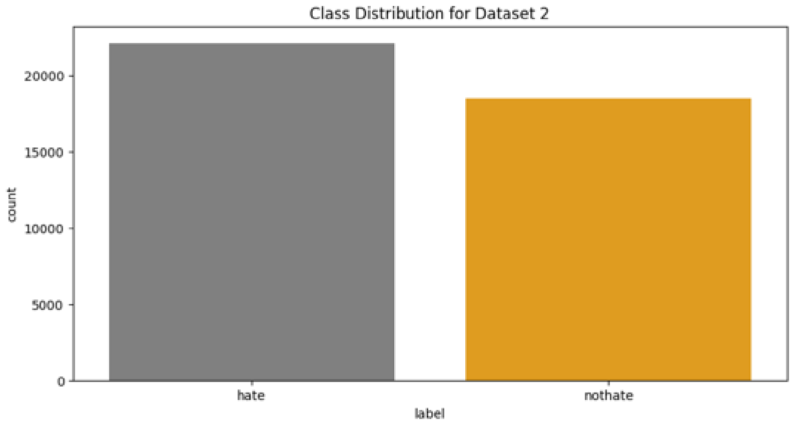
- Entry ID: Unique identifier for each entry in the dataset.
- Label: Denotes whether the entry is categorized as 'hate' or 'not hate'.
- Type: Describes the nature or category of hate speech conveyed in the entry.
- Annotator ID: Identifies the annotator responsible for labelling and annotating the entry.

This dataset mainly focuses on the words used in online content and categorizing them as either containing 'hate' or 'not hate'. It's designed to teach and test computer programs specifically made to spot and classify hate speech. The dataset contains various kinds of hate speech examples, along with labels added by experts. This helps in training computer models and analysing their performance to identify and predict hate speech accurately.

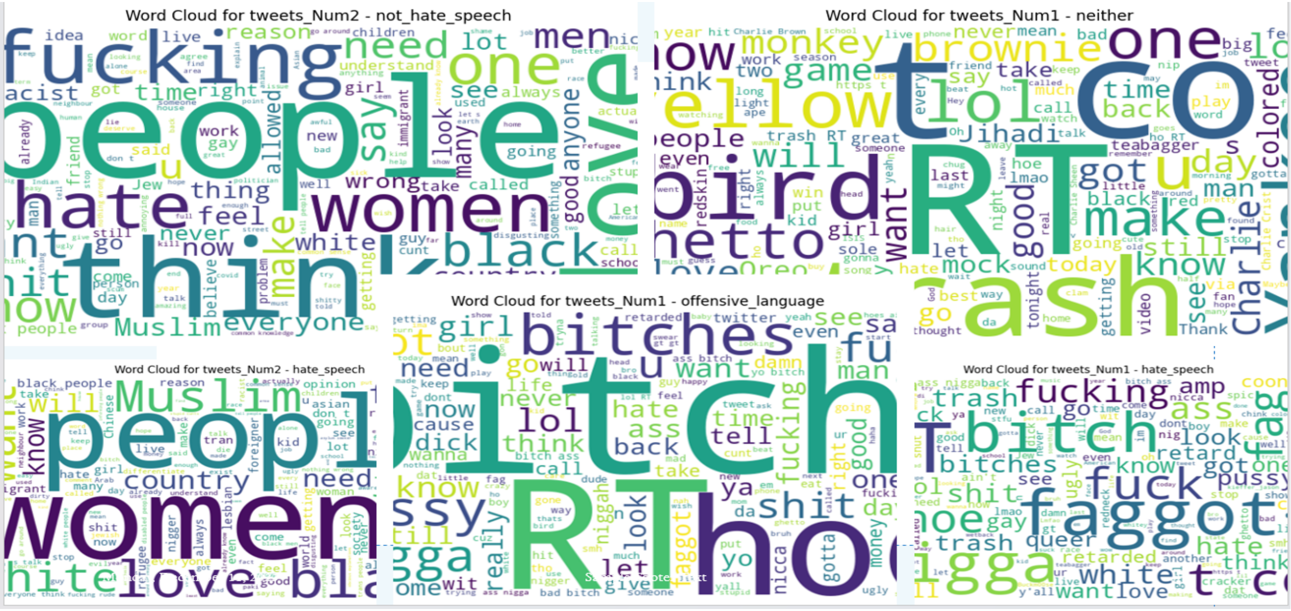
EXPLORATORY DATA ANALYSIS





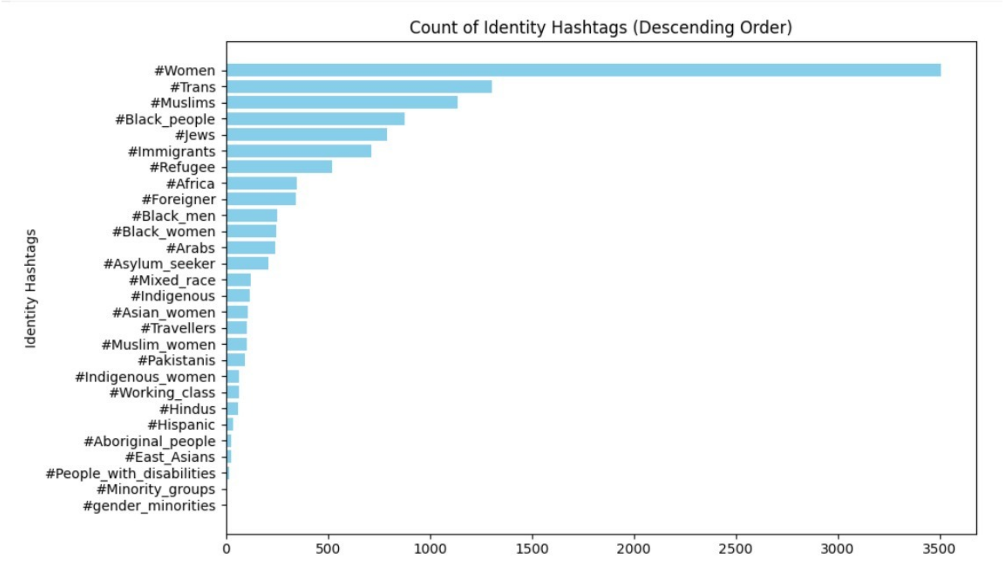


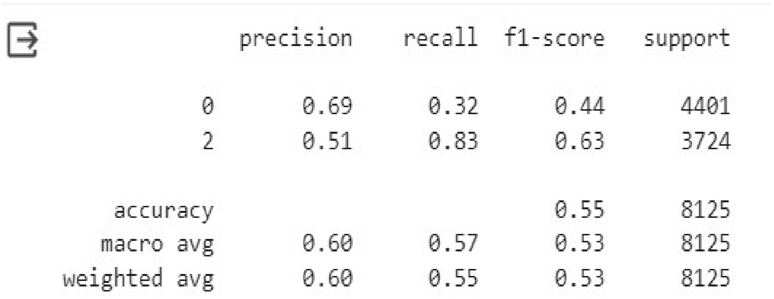
WORD CLOUD VISUALIZATIONS



ANALYSIS OF HATE HASHTAGS IN TWEET AND LOGISTIC REGRESSION MODEL

This method uses a function, 'count_hate_hashtags,' to detect hate-related hashtags linked to specific topics in tweets by employing regular expressions. The resulting dataset, 'top_hate_hashtags,' showcases tweets sorted by hate hashtag counts, offering insights into hate speech prevalence. Additionally, a visual representation of identity-related hashtags' usage rates provides valuable insights into their distribution across different social and demographic groups within the dataset.

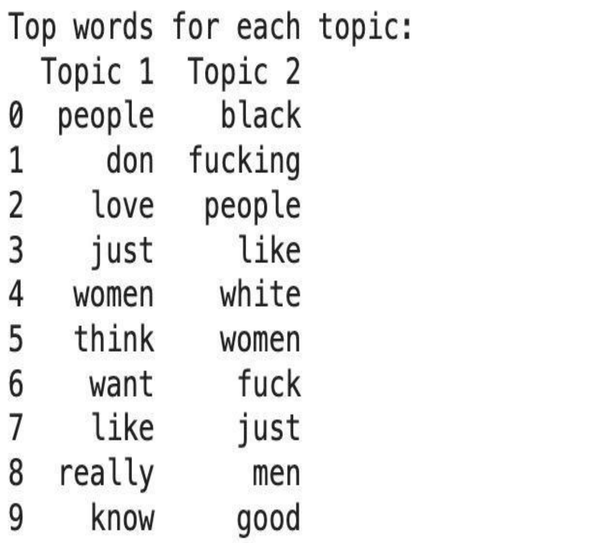




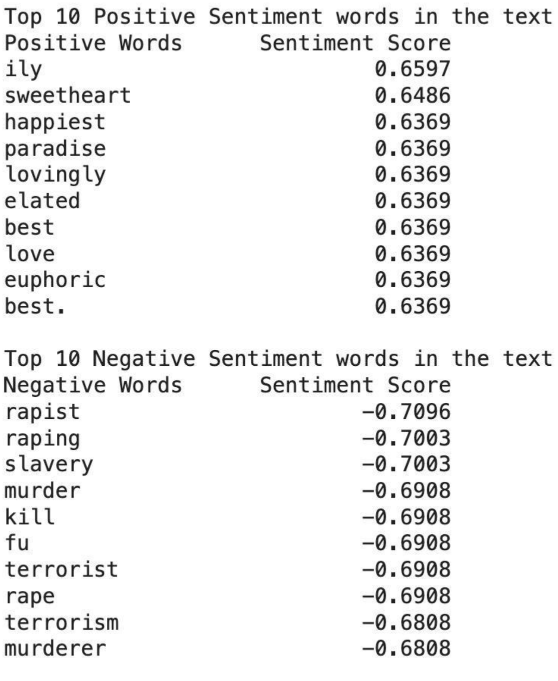
```
              precision    recall  f1-score   support

           0       0.69      0.32      0.44      4401
           2       0.51      0.83      0.63      3724

    accuracy                           0.55      8125
   macro avg       0.60      0.57      0.53      8125
weighted avg       0.60      0.55      0.53      8125
```

Moving beyond frequency analysis, it also shows the implementation of a logistic regression model utilizing word embeddings of these hashtags. The model aims to categorize hashtags into two classes—hate (class 0) and not hate (class 2). Evaluation metrics including precision, recall, f1-score, and support are furnished for both classes, shedding light on the model's performance. The F1-score is 0.44 for hate speech. It provides a balance between precision and recall, giving an overall measure of the model's accuracy in identifying hate speech. From the hashtag analysis, it is surprising, it appears that the model performs better in identifying non-hate speech compared to hate speech. The precision and recall for hate speech are relatively low, suggesting that the model struggles to accurately identify instances of hate speech, potentially indicating a need for further model optimization or data balancing techniques to improve its performance in recognizing hate speech instances. Leveraging word embeddings from these hashtags proved beneficial in identifying a higher proportion of non-hate speech instances. However, its effectiveness in recognizing hate speech was comparatively limited.

COMPARATIVE ANALYSIS OF SENTIMENT WORDS AND HATE SPEECH TOPICS



```
Top words for each topic:
  Topic 1  Topic 2
0  people    black
1     don  fucking
2    love   people
3    just     like
4   women    white
5   think    women
6    want     fuck
7    like     just
8  really      men
9    know     good
Topic distribution for the new text:
[[0.71193683 0.28806317]]
```

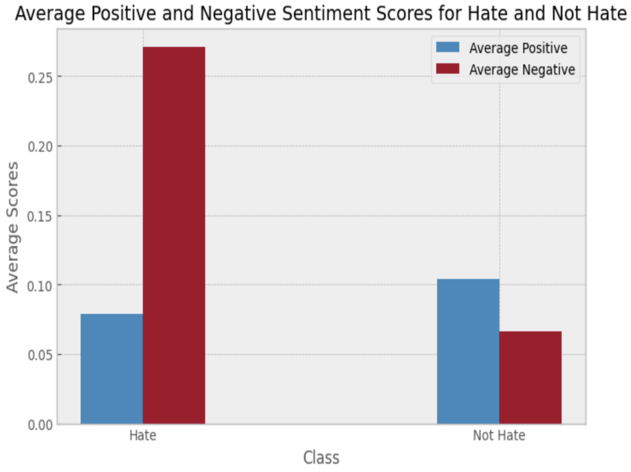


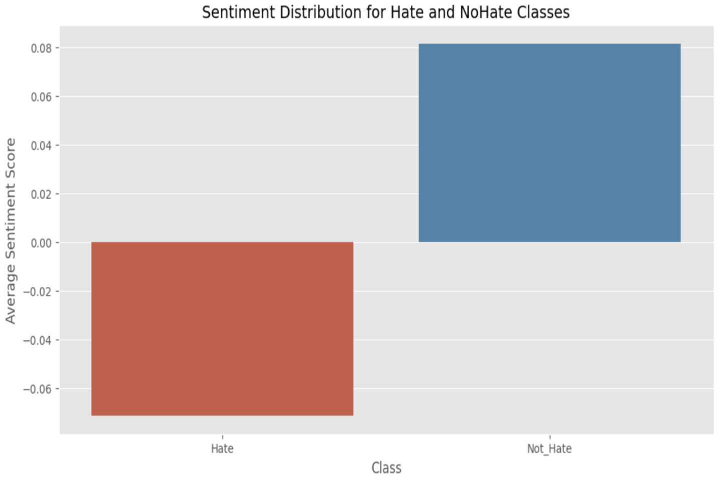
```
Top 10 Positive Sentiment words in the text
Positive Words      Sentiment Score
ily                          0.6597
sweetheart                   0.6486
happiest                     0.6369
paradise                     0.6369
lovingly                     0.6369
elated                       0.6369
best                         0.6369
love                         0.6369
euphoric                     0.6369
best.                        0.6369

Top 10 Negative Sentiment words in the text
Negative Words      Sentiment Score
rapist                      -0.7096
raping                      -0.7003
slavery                     -0.7003
murder                      -0.6908
kill                        -0.6908
fu                          -0.6908
terrorist                   -0.6908
rape                        -0.6908
terrorism                   -0.6808
murderer                    -0.6808
```

The above text presents a comprehensive text analysis focusing on sentiment and topic modelling. The left image illustrates a comparative analysis between sentiment words and hate speech topics, showcasing distinct sets of "Top words for each topic." Topic 1 encapsulates neutral or potentially positive terms like "people," "don," "love," while Topic 2 comprises explicit and negative words such as "black," "fucking," likely associated with negative sentiment or hate speech. Additionally, the "Topic distribution for the new text" numerically signifies a stronger alignment with Topic 1, highlighting the context of the new text.

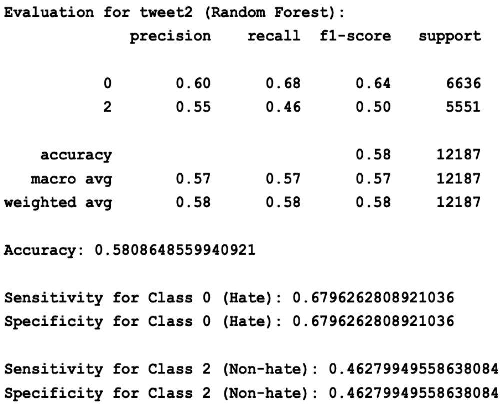
The right image features lists detailing the "Top 10 Positive Sentiment words" and "Top 10 Negative Sentiment words" extracted from the text. Positive sentiment words exhibit moderately positive scores around 0.6369, including terms like "ily," "sweetheart," while negative sentiment words, such as "rapist," "raping," showcase strongly negative scores ranging from -0.6908 to -0.7096. This analysis, pivotal in natural language processing, delves into text context, sentiment, and topics, essential for platforms monitoring social media or customer feedback. The juxtaposition of hate speech topics and profoundly negative sentiment words underlines the importance of discerning sentiment polarity and recognizing potentially harmful content, necessitating meticulous attention and moderation.

VADER SENTIMENT ANALYSIS





These charts depict the average positive and negative sentiment scores for "Hate" and "Not Hate" classes, revealing distinct emotional patterns. The "Hate" class exhibits a higher average negative sentiment score, aligning with the expected negativity in hate speech, while the "Not Hate" class showcases a greater average positive sentiment score, reflecting the positivity prevalent in non-hateful content. Visualized through bar heights representing sentiment magnitude, "Hate" reflects marked negativity, and "Not Hate" portrays positivity. This graphical representation underscores how sentiment analysis discerns between negative and positive language, aiding content moderation, social media discourse comprehension, and broader sentiment analysis across various topics.

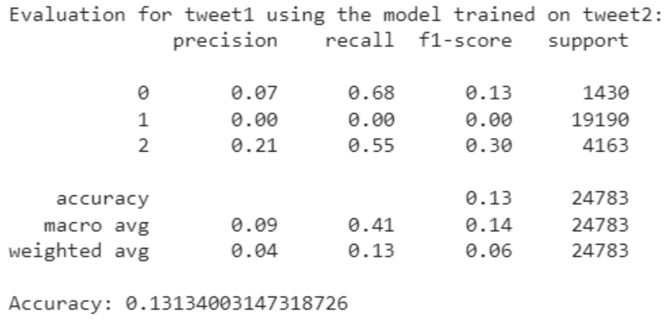


```
Evaluation for tweet2 (Random Forest):
              precision    recall  f1-score   support

           0       0.60      0.68      0.64      6636
           2       0.55      0.46      0.50      5551

    accuracy                           0.58     12187
   macro avg       0.57      0.57      0.57     12187
weighted avg       0.58      0.58      0.58     12187

Accuracy: 0.5808648559940921

Sensitivity for Class 0 (Hate): 0.6796262808921036
Specificity for Class 0 (Hate): 0.6796262808921036

Sensitivity for Class 2 (Non-hate): 0.46279949558638084
Specificity for Class 2 (Non-hate): 0.46279949558638084
```

The VADER Sentiment Analysis, integrated with a Random Forest model, was trained on sentiment features to discern between "hate" and "not hate" speech in tweets. Utilizing VADER's lexicon and rule-based approach, sentiment scores were derived per tweet, aiding classification based on positive and negative sentiment indicators. This approach enabled the model to gauge emotional polarity and intensity within language, distinguishing between hateful and non-hateful content. While the amalgamation of sentiment features facilitated the identification of negative sentiments in "hate" speech and positive sentiments in "not hate" speech, the model's overall accuracy of around 58% underscores limitations in categorizing nuanced expressions, evidenced by misclassifications in test outputs. Despite its capability in identifying prevalent negative sentiments, there's a clear need for refinement to enhance accuracy, highlighting room for improvement in discerning nuanced language nuances and boosting the model's predictive capabilities.

CROSS DOMAIN EVALUATION OF NAIVE BAYES MODEL

The assessment of a Naive Bayes machine learning model trained on "tweet2" and tested on "tweet1" data reveals deficiencies in its performance across three classes (0, 1, and 2). The reported precision, recall, and F1-score metrics portray a lackluster performance, with Class 1 exhibiting zero precision and recall, indicating an inability to make accurate predictions for this category.



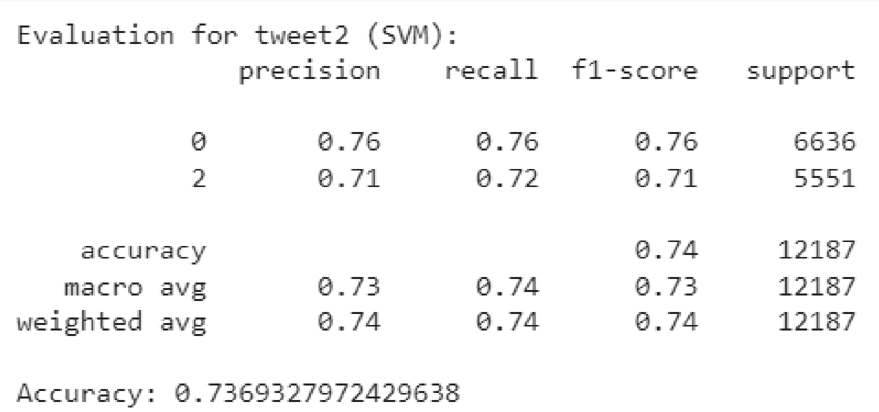
```
Evaluation for tweet1 using the model trained on tweet2:
              precision    recall  f1-score   support

           0       0.07      0.68      0.13      1430
           1       0.00      0.00      0.00     19190
           2       0.21      0.55      0.30      4163

    accuracy                           0.13     24783
   macro avg       0.09      0.41      0.14     24783
weighted avg       0.04      0.13      0.06     24783

Accuracy: 0.13134003147318726
```

Additionally, while Class 0 demonstrates a low precision of 0.07 and Class 2 displays relatively higher yet still insufficient precision and recall scores, the overall accuracy stands at approximately 0.13, signifying the model's ability to correctly identify only 13% of outcomes. Further insights from the Confusion Matrix underline misclassifications across classes, with Class 1 instances entirely misclassified and Class 2 inaccurately encompassing instances from Class 0 and Class 1. These findings underscore the model's struggle in distinguishing between classes, hinting at potential overfitting on the training data and a lack of adaptability to the test dataset.

In essence, the evaluation exposes the model's challenges in effectively discerning between classes, possibly due to overfitting on the initial training data and a limited ability to generalize to the distinct test dataset. To improve its adaptability and performance, employing transfer learning techniques could offer potential solutions, allowing the model to leverage insights from the initial training data to better understand and classify instances in new, diverse datasets like "tweet1."

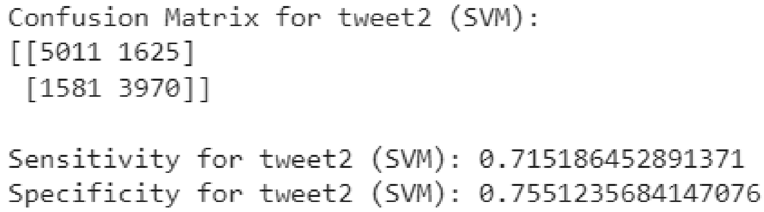
SUPPORT VECTOR MACHINE

In the examination of the Support Vector Machine (SVM) model showcased in the slide titled "SVM," an in-depth evaluation of its performance on the "tweet2" dataset is depicted. This assessment encompasses vital metrics including precision, recall, and F1-Score, indicative of the model's capability in classifying data.



```
Evaluation for tweet2 (SVM):
              precision    recall  f1-score   support

           0       0.76      0.76      0.76      6636
           2       0.71      0.72      0.71      5551

    accuracy                           0.74     12187
   macro avg       0.73      0.74      0.73     12187
weighted avg       0.74      0.74      0.74     12187

Accuracy: 0.7369327972429638
```

Notably, the SVM model demonstrates commendable precision rates of 0.76 for class 0 and 0.71 for class 2, alongside robust recall metrics of 0.76 for class 0 and 0.72 for class 2. These metrics, coupled with F1-Scores exceeding 0.70 for both classes, underline a balanced and favourable accuracy in predictions. With an overall accuracy of 0.7369, the model showcases an ability to correctly predict approximately 73.69% of the outcomes, signifying a reasonably proficient performance level.
Further dissecting the SVM model's evaluation, the slide presents a comprehensive Confusion Matrix, revealing nuanced insights into its true positive and false positive identifications for distinct classes.



```
Confusion Matrix for tweet2 (SVM):
[[5011 1625]
 [1581 3970]]

Sensitivity for tweet2 (SVM): 0.715186452891371
Specificity for tweet2 (SVM): 0.7551235684147076
```

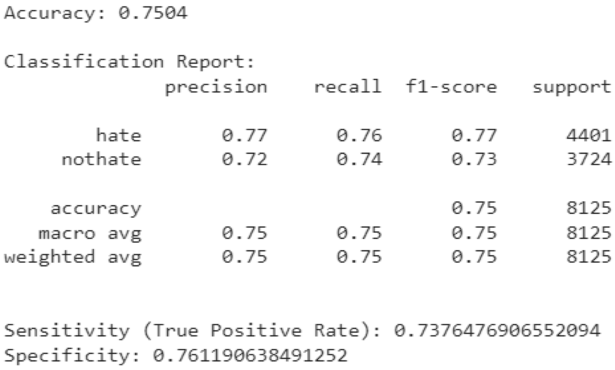
While indicating 5011 true positives for class 0 and 3970 true positives for class 2, it also highlights 1625 false positives for class 0 and 1581 false negatives for class 2. These findings suggest areas for refinement in minimizing misclassifications, particularly false positives and negatives, to bolster the model's precision and recall. Additionally, the inclusion of sensitivity and specificity metrics, portraying a sensitivity of 0.7152 and a specificity of 0.7551, underscores a balanced identification of true positives and true negatives within the "tweet2" dataset, solidifying the model's reasonably balanced performance despite areas for enhancement.

BERT

BERT is particularly effective for identifying context in text due to its bidirectional architecture and deep contextualized embeddings. Unlike traditional models that read text in one direction (either left to right or right to left), BERT utilizes a bidirectional transformer architecture. It processes the entire input sequence in both directions during training. This enables the model to understand the context of a word by considering all the words in a sentence, capturing dependencies and relationships between them. The BERT-based hate speech classification model demonstrated robust performance on Twitter data, achieving a notable accuracy and effectively distinguishing between hate speech and non-hate speech instances. The training process, guided by binary cross-entropy loss and Adam optimization, contributed to the model's ability to capture intricate patterns in the language of tweets. Leveraging the power of BERT embeddings, the model exhibited a nuanced understanding of context, enabling it to make informed predictions.

MULTI-LAYER PERCEPTRON

The Multi-Layer Perceptron (MLP) neural network model, tailored for discerning "hate" and "not hate" categories in tweet data, showcases a sophisticated architecture. It features an intricate structure with an input layer containing 1739 nodes using Sigmoid activation, progressing through three hidden layers of 200, 140, and 70 nodes employing Rectified Linear Unit (ReLU) functions, concluding in an output layer with 2 nodes activated by Softmax. Developing a Multi-Layer Perceptron (MLP) model involves several steps. Initially, the text is transformed into a TF-IDF feature matrix and a Part-of-Speech (POS) TF matrix. Additional linguistic features like syllable count, average syllables per word, and the number of unique words are derived. Next, Logistic Regression with L1 regularization is utilized to identify the most crucial features. The selected feature vector is then fed into an MLP Network for classification. The output layer employs softmax activation for classifying into either 3 or 4 classes.

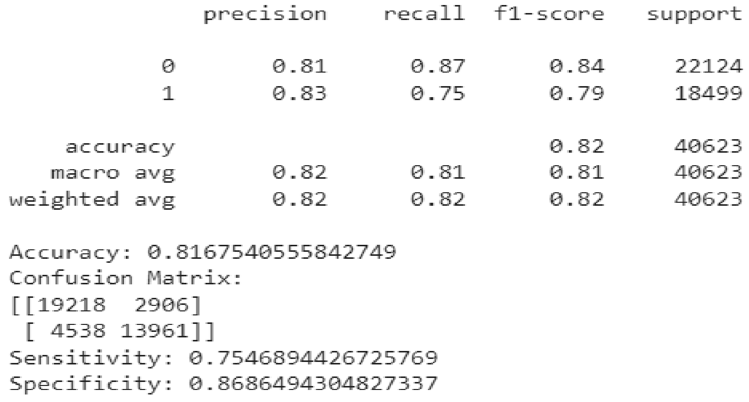


```
Accuracy: 0.7504

Classification Report:
              precision    recall  f1-score   support

        hate       0.77      0.76      0.77      4401
     nothate       0.72      0.74      0.73      3724

    accuracy                           0.75      8125
   macro avg       0.75      0.75      0.75      8125
weighted avg       0.75      0.75      0.75      8125


Sensitivity (True Positive Rate): 0.7376476906552094
Specificity: 0.761190638491252
```

The model's evaluation highlights its performance through various metrics: achieving an accuracy of 0.7504, demonstrating a 75.04% accuracy in test data prediction, balanced precision (0.77), recall, and F1-Score for the "hate" class, indicating adeptness in identifying hate speech instances. The "not hate" class shows slightly lower but comparable metrics, signifying a balanced yet imperfect classification. Additionally, with sensitivity at approximately 0.7376 and specificity around 0.7611, the model showcases competence in correctly identifying instances of both "hate" and "not hate." Despite its notable accuracy and balanced metrics, the model exhibits instances of misclassifications, suggesting avenues for enhancement, particularly in reducing false positives and negatives to further improve overall accuracy.

CNN + GLOVE EMBEDDING

The first model, a Convolutional Neural Network (CNN) likely trained on tweet data, boasts a sequential architecture involving embedding layers, convolutional layers, global max pooling, dropout, and dense layers. Constructing a Convolutional Neural Network (CNN) for hate speech detection involves several steps. First, a Word2Vec conversion is used to create a 300-dimensional word embedding GloVe model, which is pre-trained. The embedding dimension is set to 100 * 300. This embedding is then fed into the DCNN model for classification. The CNN architecture consists of four Conv1D layers, each with 300 filters and window sizes of 1, 2, 3, and 4. K-max pooling is applied correspondingly to each Conv1D layer, and the outputs are merged into a single vector. This merged vector is passed through Dropout, a dense layer, and a softmax layer for classification.



```
              precision    recall  f1-score   support

           0       0.81      0.87      0.84     22124
           1       0.83      0.75      0.79     18499

    accuracy                           0.82     40623
   macro avg       0.82      0.81      0.81     40623
weighted avg       0.82      0.82      0.82     40623

Accuracy: 0.8167540555842749
Confusion Matrix:
[[19218  2906]
 [ 4538 13961]]
Sensitivity: 0.7546894426725769
Specificity: 0.8686494304827337
```

With 1,860,529 parameters, it achieves an accuracy of approximately 82%, exhibiting strong precision and recall for both classes, albeit with a slight bias towards higher recall for class 0 and higher precision for class 1. The confusion matrix indicates distinct true positives, false negatives, false positives, and true negatives, with sensitivity around 0.75 and specificity at approximately 0.87, showcasing its effectiveness for classification tasks.

Additionally, the "CNN + GloVe" sentiment analysis model swiftly processes tweets, providing sentiment predictions through GloVe word embeddings in just 81 milliseconds per step. While excelling in straightforward sentiment identification, its occasional misclassifications hint at challenges in handling nuanced language nuances and contextual complexities. These models showcase remarkable capabilities in discerning sentiment in text data, despite encountering complexities inherent in language analysis.

CONTEXTUAL ANALYSIS

From the different explorations of our models, we explored different embedding that behaved differently. Out of the different ones, GloVe embedding worked the best for our scenario. GloVe embedding captures the semantic relationships between words based on their co-occurrence in a large corpus. This semantic richness allows the model to understand the contextual nuances of language, crucial for distinguishing hate speech from non-hateful content on Twitter. This provides a continuous vector representation for each word, enabling the CNN model to grasp the subtle variations in language, enhancing its ability to discern hate speech patterns within the dataset. GloVe embeddings, by capturing the context in which words appear, help resolve polysemy. This is particularly valuable in the context of social media where phrases can have diverse interpretations.

ETHICAL CONSIDERATIONS

Freedom of Speech vs Hate Speech: This consideration highlights the need to distinguish between protecting freedom of speech and preventing hate speech. When implementing AI systems for moderating content, it's crucial to avoid over-censoring, which could suppress legitimate expression.

Bias & Fairness: Fairness in AI systems is about ensuring that the algorithms are impartial and do not discriminate against any individual or group. Mitigating biases in machine learning models is essential to avoid discriminatory outcomes and unintended consequences that could arise from skewed data or biased algorithms.

Robustness & Security: Robustness refers to the model’s ability to handle diverse and challenging inputs without failure. Security involves the model's resilience against attempts to deceive or manipulate it. Ensuring that a model is not easily fooled by users or administrators is essential for maintaining the integrity of its functionality.

In all, these considerations stress the importance of ethical development and deployment of AI systems, where balance, impartiality, and resilience are crucial to the beneficial application of technology in society.

CONCLUSION

Robust Model Construction: Prioritizing the development of models that are robust in order to mitigate biases. Robustness in this context refers to the model's ability to perform accurately and fairly under a variety of conditions and to resist being influenced by biases in the data.
It also mentions the importance of addressing concerns related to the unintentional amplification of stereotypes or discriminatory behaviour by users. This means the model should not reinforce negative stereotypes or discriminatory patterns present in the training data.

Real-time Monitoring Integration:There is an emphasis on designing models that can be integrated seamlessly into real-time monitoring systems. This implies that the model should be capable of processing data and making predictions or decisions in a live environment, without significant delays.

The need for swift detection and response to potential instances of hate speech as they occur. This indicates that the model should not only identify hate speech accurately but also trigger an appropriate response quickly to prevent harm or the spread of such content.

In summary, the slide is advocating for the ethical development of AI models that are not only technically competent but also considerate of social implications, ensuring fairness, preventing discrimination, and functioning effectively in real-world applications where immediate responses are necessary.

REFERENCES

1. https://www.researchgate.net/publication/363212595_Hate_Speech_Detection_Using_Text_Mining_and_Machine_Learning
2. https://aclanthology.org/2021.acl-long.132.pdf
3. https://www.sciencedirect.com/science/article/pii/S0925231223003557
4. https://www.kaggle.com/code/giovanimachado/hate-speech-bert-cnn-and-bert-mlp-in-tensorflow
5. https://towardsdatascience.com/social-media-sentiment-analysis-with-vader-c29d0c96fa90

“We're using ChatGPT to generate definition of contents and train models for identifying hate speech. Our aim is to improve the accuracy of hate speech detection to create a safer online environment”