INFOSYS SPRINGBOARD INTERNSHIP

HATE SPEECH DETECTION IN TWITTER PLATFORM

BY VAISHNAVI SAMAL

GROUP 2

CONTENT:

1. BUSINESS PROBLEM:

In today's digital age, hate speech on online platforms is a growing concern that impacts individuals and communities globally. Social media platform like Twitter host billions of users who generate a vast amount of content daily. Despite the benefits of these platforms, the prevalence of hate speech poses significant challenges for both users and platform providers.

- **Challenges:**

  1. **Psychological Harm:** Hate speech can cause significant emotional and psychological distress to targeted individuals and groups.
  2. **Social Unrest:** It has the potential to incite violence, perpetuate discrimination, and destabilize communities.
  3. **Legal Risks:** Platforms face increasing legal scrutiny and potential penalties for failing to manage hate speech effectively.
  4. Brand Reputation: Inadequate management of hate speech can severely damage the platform’s reputation and credibility.

2. PROPOSED SOLUTION:

To address the problem of hate speech on Twitter platform, we propose developing a machine learning model that can automatically detect and categorize user-generated content into hate speech/offensive language and neutral content. The solution involves the following steps:

1. **Data Collection:** Gather a diverse and comprehensive dataset of user-generated content from Twitter platforms.
2. **Data Labeling:** Ensure accurate labeling of content into the categories of hate speech/offensive language, or neutral.
3. **Model Training:** Use the labeled dataset to train a robust machine learning model.
4. **Evaluation:** Test and validate the model to ensure high accuracy and reliability in detecting hate speech.
5. **Deployment:** Integrate the model into online platforms to assist in real-time content moderation.

3. DATASET DESCRIPTION:

After gone through various datasets, we selected Davidson dataset that best met our criteria for comprehensiveness, diversity, and quality. The key reasons for our choice include:

1. **Comprehensive and Representative:**
    - **Extensive Coverage:** Over 24,000 entries, covering a wide range of hate speech scenarios and user-generated content from Twitter platform.
2. **Quality and Accuracy:**
    - **Rigorous Annotation:** Multiple annotators from diverse backgrounds reviewed and labeled each entry, ensuring high accuracy and consistency.
3. **Relevance and Impact:**
    - **Operational Efficiency:** Enhances user experience by reducing exposure to hate speech, potentially increasing user engagement and retention, thus boosting platform revenue and reducing the risk of legal penalties.

4. **Labeled Categories:**

- Each tweet is labeled as hate speech, offensive language, or neutral content.

5. **Dataset Structure:**

- **count:** Number of occurrences/interactions of the tweet.
- **hate_speech:** Indicates the presence of hate speech in the tweet.
- **offensive_language:** Indicates the presence of offensive language in the tweet.
- **neither:** Indicates the tweet does not contain hate speech or offensive language.
- **class:** Categorization of the tweet (2 = neither, 1 = offensive language, 0 = hate speech).
- **tweet:** The actual text content of the tweet.

6. **Example :**

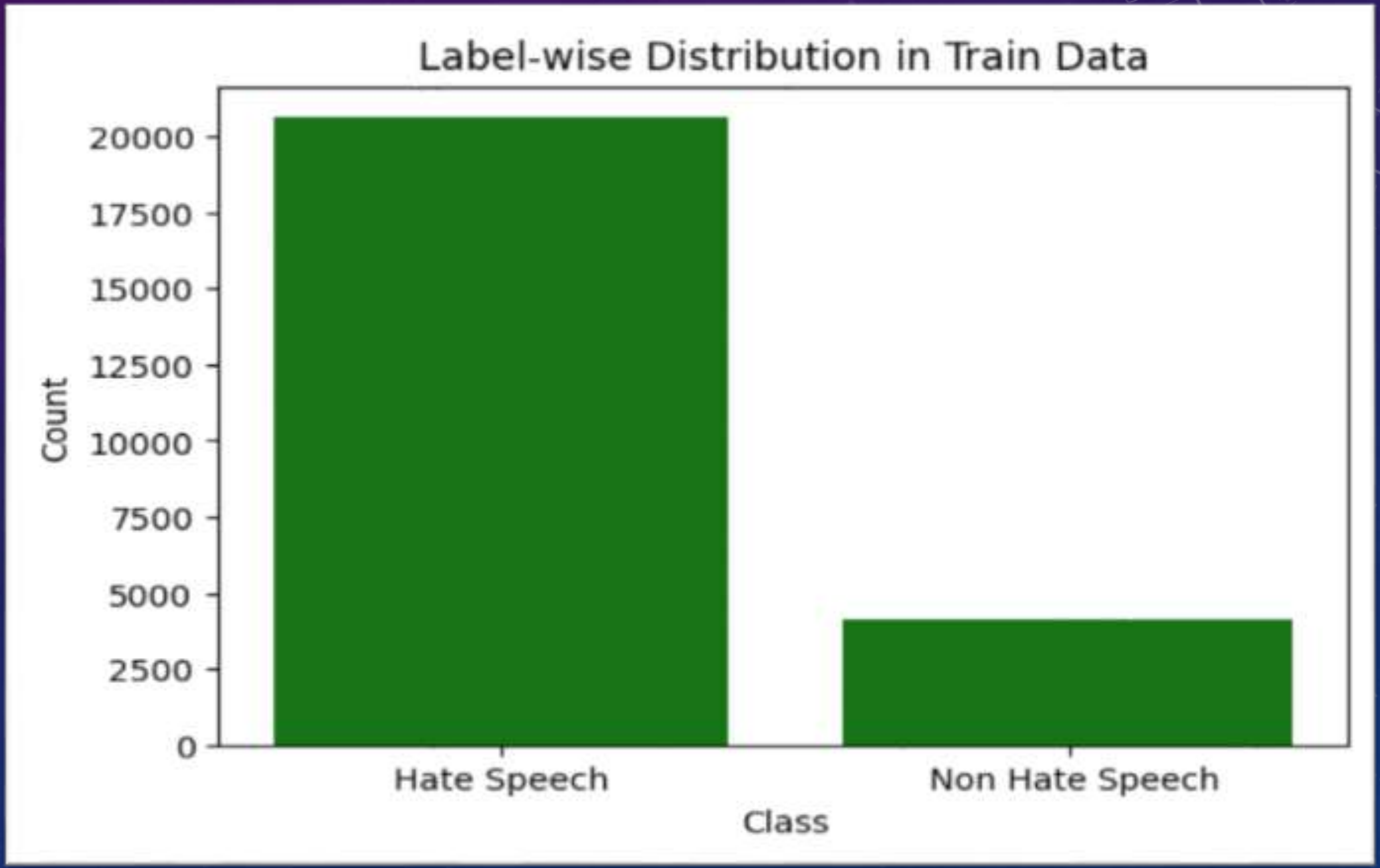
count	hate_speech	offensive_language	neither	class	tweet
3	0	3	0	1	" Murda Gang bitch its Gang Land "
3	0	2	1	1	" So hoes that smoke are losers ? " yea ... go on IG

4. DATA DISTRIBUTION

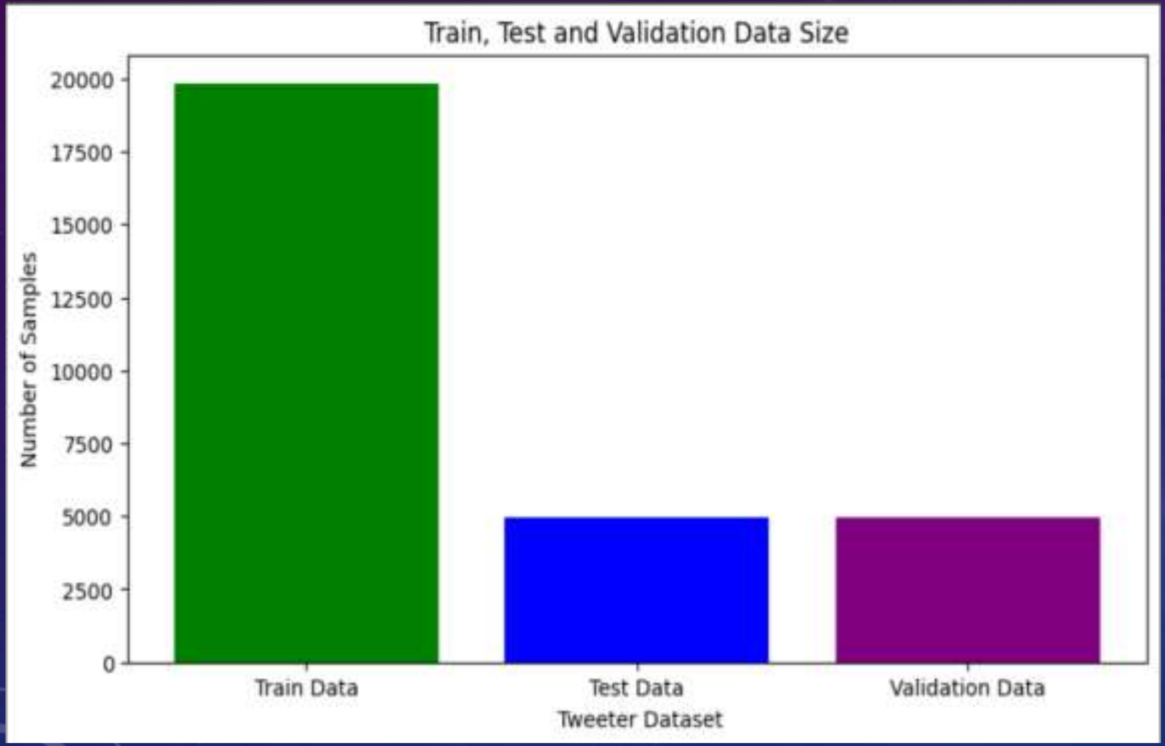
Label-wise counts in Data:

Hate speech -20609

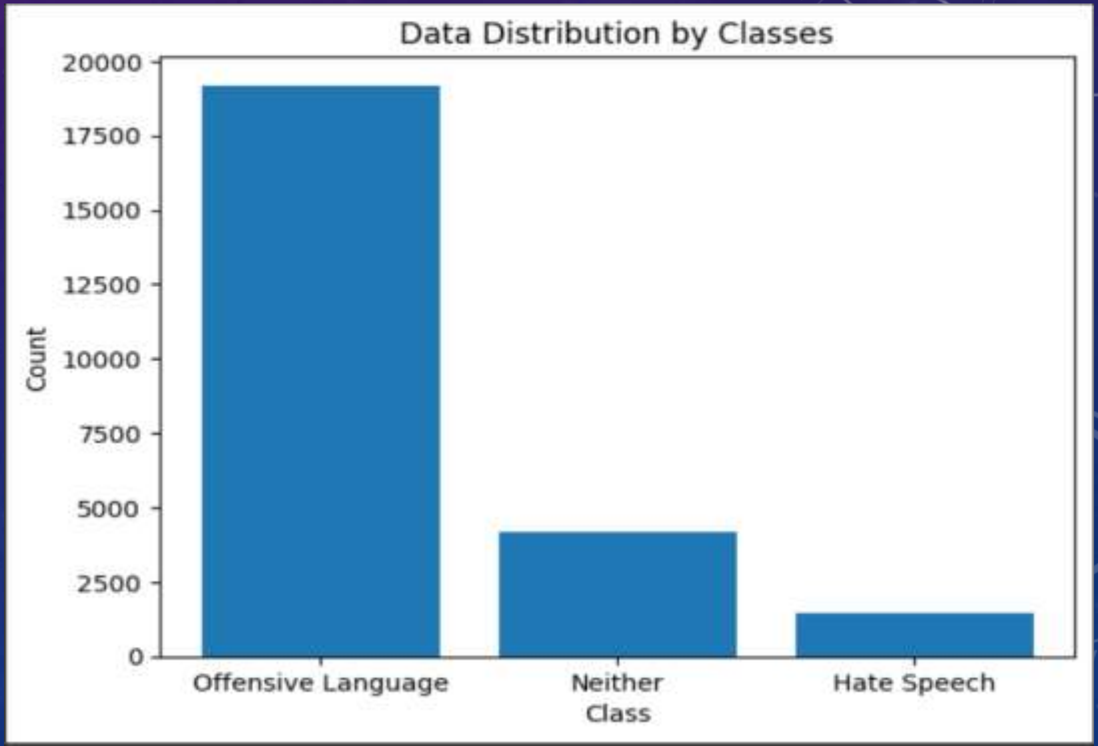
Non Hate speech- 4159



- **Train, Test and Validation Data Size:**



- **Class wise Data Distribution :**



5. DATA PREPROCESSING

- Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.
- Steps included for Data Preprocessing :

1. **Recategorization:** Labeled Hate speech and offensive language as 0, non-hate speech as 1
2. **Handling missing values:** Replaced with empty strings
3. **Handling duplicates:** Removed duplicates
4. **Handling abbreviations:** Replaced with full forms
5. **HTML entity decoding:** Decoded HTML entities
6. **Contraction expansion:** Expanded contractions
7. **Normalization:** Applied normalization to the data

6. TOKENIZATION AND EMBEDDING TECHNIQUES

- **Tokenization** : **Word Tokenization**

Word tokenization divides the text into individual words. In this tokenization technique, words are treated as the basic units of meaning.

- **Embedding techniques :**

1. **One – Hot Encoding:** One hot encoding is one method of converting data to prepare it for an algorithm and get a better prediction. With one-hot, we convert each categorical value into a new categorical column and assign a binary value of 1 or 0 to those columns. Each integer value is represented as a binary vector.
2. **TF – IDF Encoding :** TF-IDF is a numerical statistic that reflects the importance of a word in a document. The TF-IDF algorithm takes into account two main factors: the frequency of a word in a document (TF) and the frequency of the word across all documents in the corpus (IDF).
3. **Word2Vec Encoding :** Word2Vec builds word vectors, which are distributed numerical representations of word features. These word features may include words that indicate the context of the specific vocabulary words present individually.

7. MODELING

- **Machine Learning Model:**

1. Random Forest Model
2. Naive Bayes Model
3. Logistic Regression Model

- **Deep Learning Models:**

1. Artificial Neural Network (ANN)
2. Convolutional Neural Networks (CNN)

- **Finalized Deep Learning Model : Convolutional Neural Networks (CNN)**

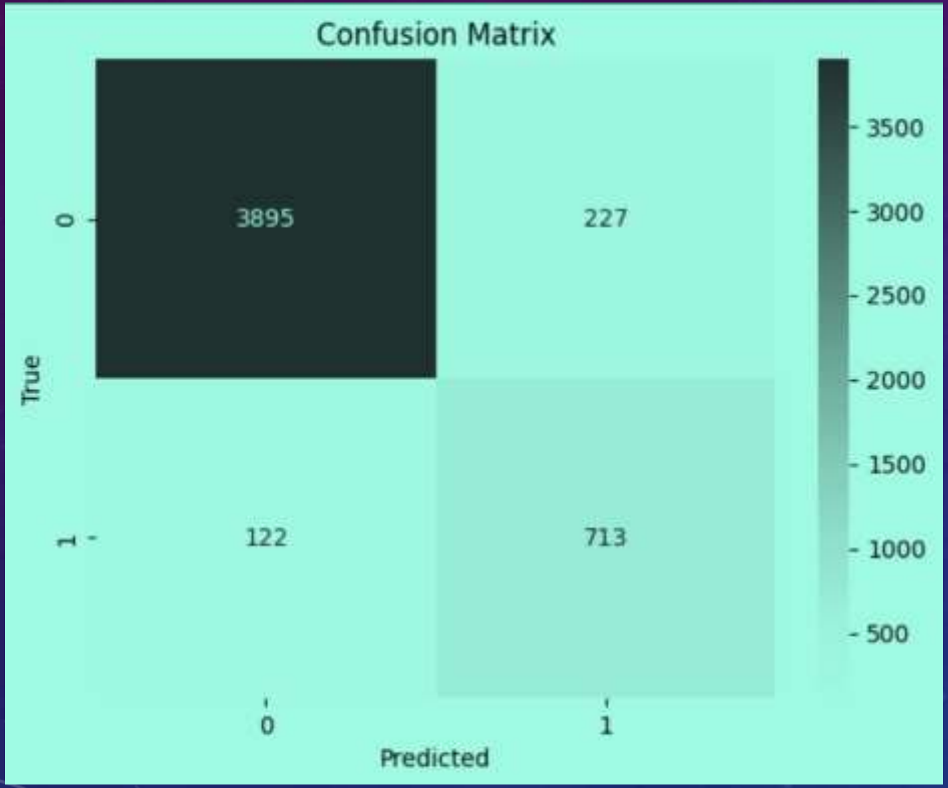
The architecture of CNNs is inspired by the visual processing in the human brain, and they are well-suited for capturing hierarchical patterns and spatial dependencies.

- Reasons for choosing Convolutional Neural Networks Model:
  - CNNs are relatively robust to noise and variations in the input data.
  - CNNs can be adapted to a variety of different tasks by simply changing the architecture of the network.
  - CNNs can be very efficient, especially when implemented on specialized hardware such as GPUs.

Confusion Matrix of CNN Model

- **Classification Report of CNN Model:**

- **Confusion Matrix of CNN Model :**



Accuracy: 0.9295945128101675
Classification Report:

	precision	recall	f1-score	support
0	0.97	0.94	0.96	4122
1	0.76	0.85	0.80	835
accuracy			0.93	4957
macro avg	0.86	0.90	0.88	4957
weighted avg	0.93	0.93	0.93	4957

8. EVALUATION METRICS

- **Key Matrix for Evaluation: F1 Score**

The F1 score is the harmonic mean of precision and recall, providing a single metric to assess the balance between the two.

- **Why F1 Score Over Other Parameters:**
  - Balance of Precision and Recall: The F1 score strikes a balance between precision and recall, making it suitable for tasks where both false positives and false negatives have significant consequences. In hate speech detection, misidentifying non-hate speech tweets as hate speech (false positives) or failing to identify hate speech tweets (false negatives) can impact the effectiveness of content moderation.
  - Suitability for Imbalanced Datasets: In Tweeter datasets, hate speech instances may be rare compared to non-hate speech content. The F1 score's harmonic mean ensures that both types of errors are equally penalized, providing a fair assessment of model performance across classes.
- F1 Scores Achieved Using **Convolutional Neural Networks (CNN):**
  - Class 0: 0.96
  - Class 1: 0.80

THANK YOU