INFOSYS SPRINGBOARD INTERNSHIP

TOPIC: HATE SPEECH DETECTION ON TLEGRAM PLATFORM

Business Problem:

Hate speech on online platforms is a growing concern, especially on messaging apps like Telegram, where content can spread quickly and often without moderation. Hate speech can cause significant harm to individuals and communities, leading to psychological distress, social division, and, in some cases, real-world violence. For businesses, the presence of hate speech on their platforms can damage their reputation, lead to legal liabilities, and result in user attrition. Telegram must implement an effective hate speech detection system to maintain a safe and welcoming environment for all users, uphold its brand image, and comply with regulatory standards.

SOLUTION:

To address the problem of hate speech on Telegram, we are leveraging the power of using BERT tokenization and embeddings in combination with a deep learning model featuring three BiLSTM layers. Our approach begins with collecting a comprehensive dataset of Telegram messages that are labeled as either hate speech or non-hate speech. We then clean and preprocess this text data to remove special characters, handle emojis, and normalize the text for better processing. The core of our solution involves using the BERT tokenizer to convert these messages into token IDs, These tokenized messages are then passed through a pre-trained BERT model to obtain rich, contextual embeddings that capture the semantic relationships between the words in each message. These BERT embeddings are subsequently fed into a neural network comprising three Bidirectional Long Short-Term Memory (BiLSTM) layers. These layers are adept at capturing long-range dependencies in text data from both forward and backward directions, allowing our model to understand the nuanced patterns and context in the messages more effectively.

DATASET DESCRIPTION:

The GHC (Gab Hate Corpus) dataset is a comprehensive collection of posts from Gab, a social media platform known for its lenient moderation policies, which has been associated with a high incidence of hate speech. The dataset is specifically curated for hate speech detection research and contains a total of 22,037 posts. Each post is meticulously annotated, not only for the presence of hate speech but also with additional columns that provide valuable insights into different aspects of the content. These columns include 'hd' for Human Dignity, 'cv' for Calls for Violence, and 'vo' for Vulgarity/Offensive Language directed at an individual. This detailed annotation makes the GHC dataset an essential resource for developing and refining automated systems to identify and mitigate hate speech in online platforms.

Example:

1).Text: "he most likely converted to islam due to his nature being suitable for islamic doctrine prophet muhammad was a psychopath"
• hd: 0 • cv: 0 • vo: 0 • label: 0 (No hate speech)

2).Text: "italy candidate says that all immigrants cannot be accepted if we want our white race our society to continue denounces nonwhite immigration as invasion closetheborder "
• hd: 1 • cv: 0 • vo: 0 • label: 1(Hate speech)

3).Text: "men have always gone to war to protect the women and children if the women are fat feminist toads and our children are a bunch of tranny shit turds then rest assured the military will dry the fuck up"
• hd: 1 • cv: 0 • vo: 1 • label: 1(Hate speech)

Data Distribution:

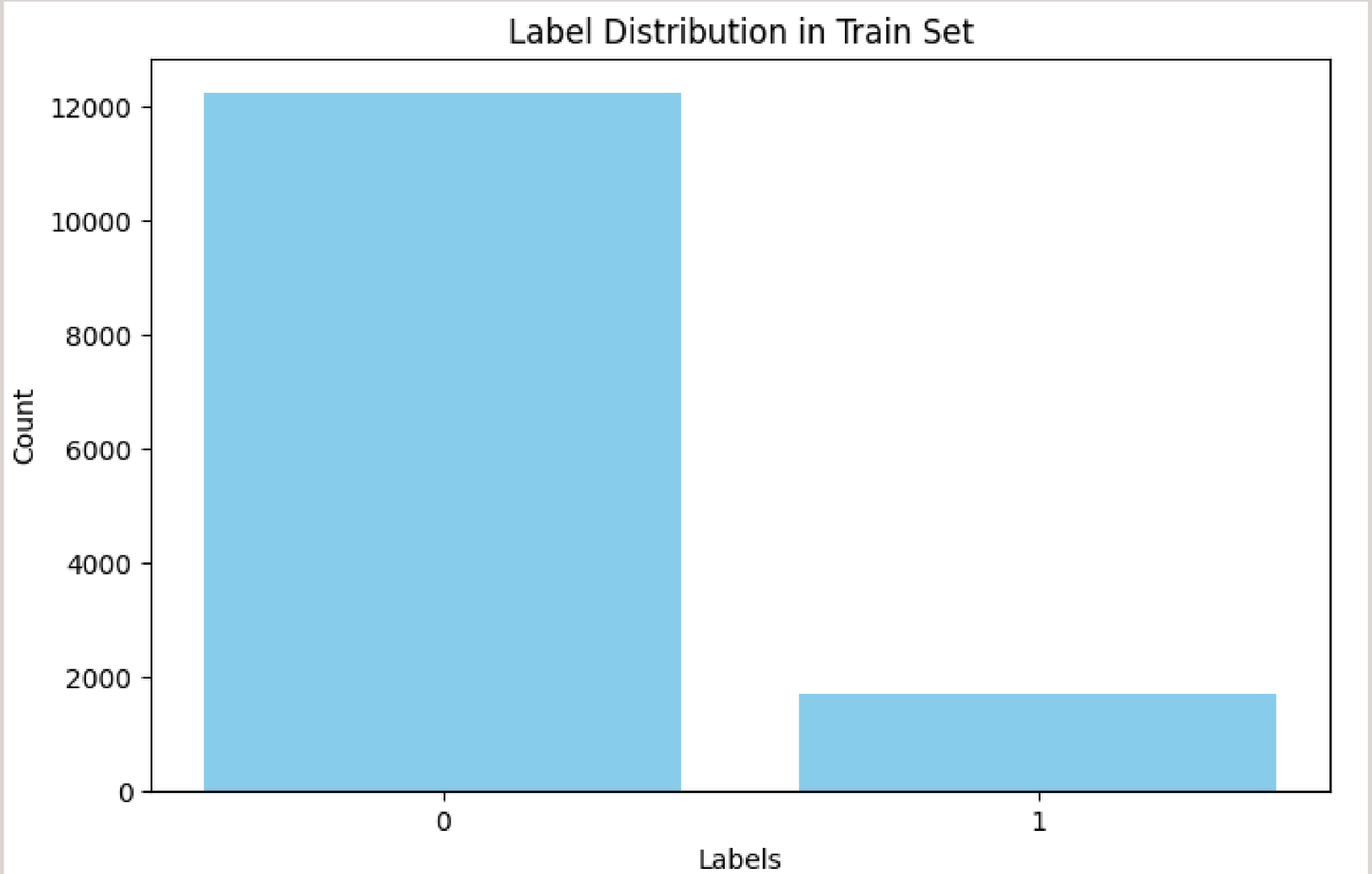
The dataset is specifically curated for hate speech detection and contains a total of 22,037 posts

Label wise splitting:

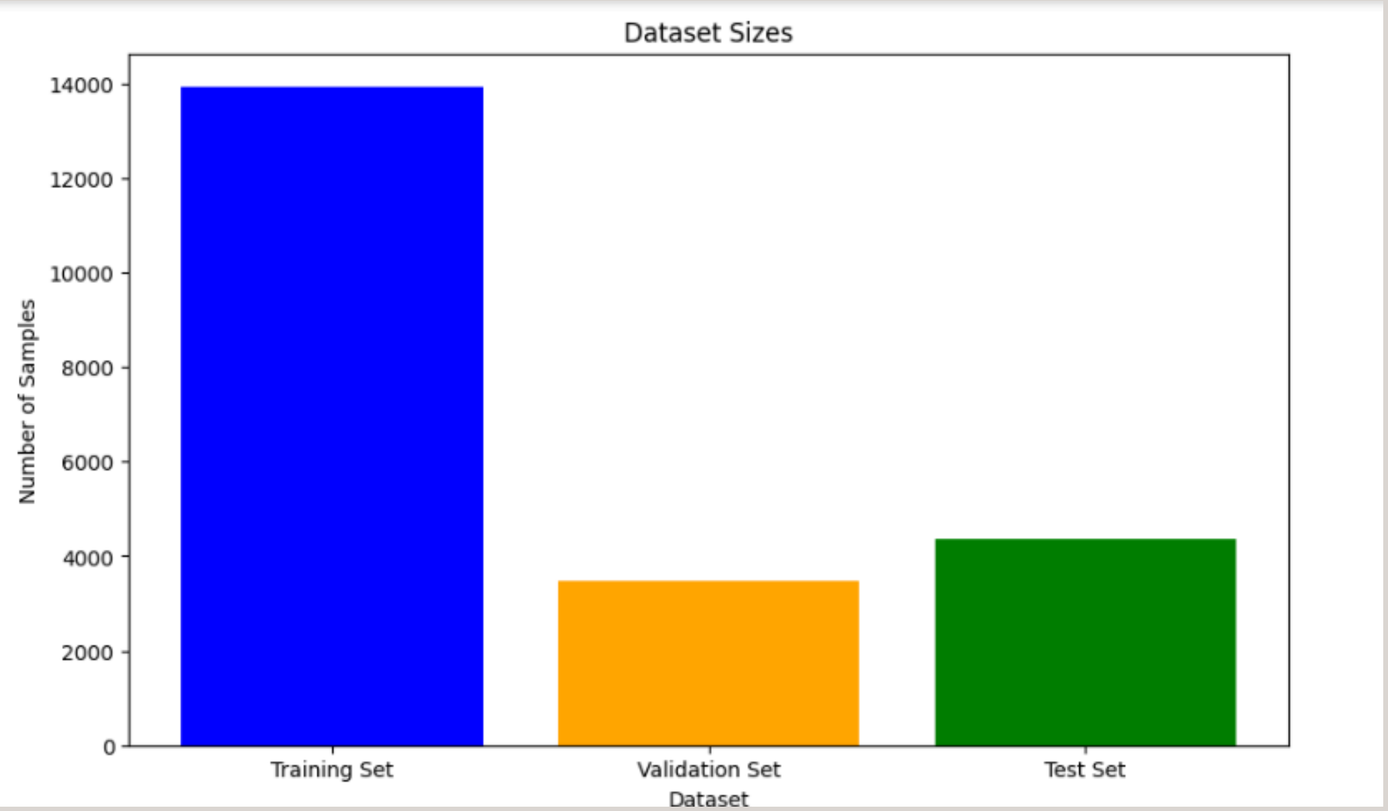
Label:-

0: 12240

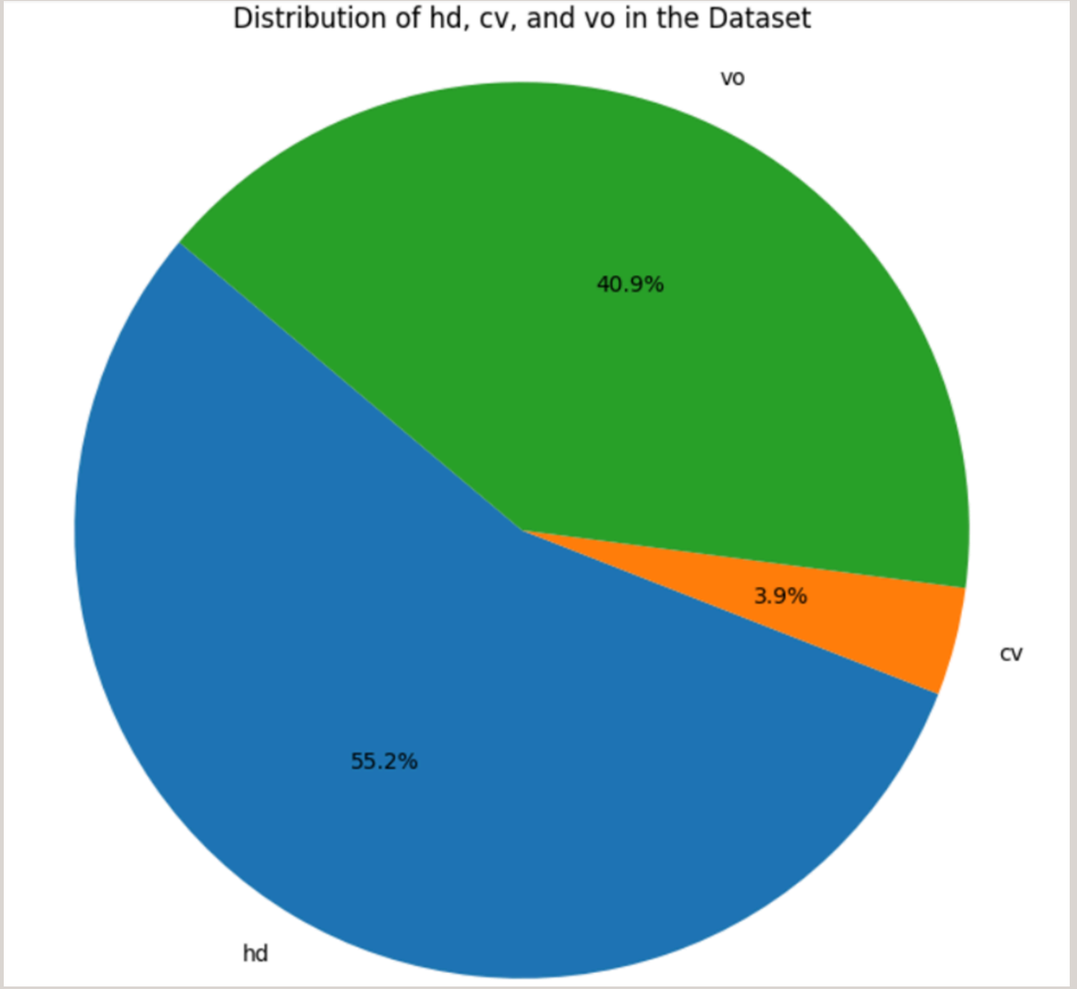
1 : 1696



- **Train,test,validation distribution:**



- **Metadata distribution**



DATA PREPROCESSING:

1. **Lowercasing**: Converting all text to lowercase ensures uniformity and reduces the complexity caused by case sensitivity.

2.**Punctuation Removal:** Punctuation often does not add value in hate speech detection and can be removed to simplify the text.

3. **Stopword Removal**: Stopwords are common words (like "and", "the", "is") that do not carry significant meaning and are often removed to focus on more meaningful words.

4.**Stemming**: Stemming reduces words to their root form. For example, "running" and "runner" are both reduced to "run"

5. **Lemmatization:** Lemmatization is similar to stemming but more accurate as it reduces words to their meaningful base form. For example, "better" is reduced to "good".

6. Removing URLs, Hastags , & mentions: Hashtags, and mentions (e.g., @username) are often irrelevant for hate speech detection and can be removed to clean the text.

7.Emoji Conversion: Emojis can carry sentiment and meaning, so converting them to text can help preserve their information.

8.Remove extra space: Trimming excess whitespace between words to maintain uniform spacing and improve readability.

TOKENISATION & EMBEDDING

For Machine Leaning:

Tokeniser:Whitespace Tokeniser:

With the help of whitespace tokeniser we get tokens from string of words or sentences without whitespaces, new line and tabs.

Embedding

1.One Hot Encodding: One-hot encoding is a technique used to convert categorical variables (like class labels) into a format that can be provided to machine learning algorithms to improve model performance.

2.Label Encodding:Label encoding is another technique used to convert categorical variables into numerical format. Unlike one-hot encoding, which creates binary columns for each category, label encoding assigns a unique integer to each category.

3.Word2vec Embedding: It represents words in a continuous vector space where words with similar meanings have similar representations.Word2Vec embeddings are powerful tools for capturing semantic meanings from text data

4. TF-IDF Embeddings: The values in TF-IDF vectors have clear meanings.Term Frequency (TF) measures the frequency of a term in a document, while Inverse Document Frequency (IDF) measures how unique or rare a term is across all documents. This helps in understanding the importance of terms within and across documents. It helps in highlighting important words in a document while reducing the impact of less informative, common words (e.g., "the", "is", "and").

For Deep learning Models:

BERT Tokeniser:

- BERT uses a method called WordPiece tokenization. This method breaks words into subword units. For example, the word "unhappiness" might be tokenized into "un", "happiness".
- The tokenizer creates attention masks to indicate which tokens are actual words and which are padding. This helps the model differentiate between real content and padding during processing.

BERT Embeddings:

- BERT (Bidirectional Encoder Representations from Transformers) embeddings are a type of word embedding generated by the BERT model, which captures rich contextual information from both directions (left-to-right and right-to-left) in the text.

Modeling:

ML Models:

1. Logistic Regression
2. Random Forest
3. Naive Bayes(Multinomial)
4. SVM(Support Vector Machine)
5. Gradient Boosting

Finalised Model: Gradient Boosting

Reason:

- This technique combines the predictions of multiple weak learners, typically decision trees, in a sequential manner. Each subsequent tree in the ensemble corrects the errors of its predecessor, gradually improving the overall model's accuracy.
- iterative process makes Gradient Boosting particularly adept at capturing complex patterns and relationships within the data, which is crucial in distinguishing hate speech from non-hate speech.

DL Model:

1.CNN
2.Bideirectional LSTM
3.CNN with LSTM
4.Bidirectional LSTM with BERT Embeddings

Finalised Model: Bidirectional LSTM with BERT Embeddings

Reason: Combining Bidirectional Long Short-Term Memory (BiLSTM) networks with BERT embeddings provides a powerful approach for hate speech detection. BERT (Bidirectional Encoder Representations from Transformers) embeddings offer deep contextualized representations of text, while BiLSTMs excel at capturing sequential dependencies in both forward and backward directions. This combination can significantly enhance the model's ability to understand the context and nuances in text data, which is crucial for accurately identifying hate speech.

EVALUATION METRICS:-

Recall:

Recall measures the effectiveness of a classification model in identifying all relevant instances from a dataset.

- **For ML Model:** 58%
- **For DL Model**: 78%

ROC AUC Score:

AUC measures how well a model is able to distinguish between classes.

- **For ML Model**: 79%
- **For DL Model:** 87%

THANK YOU

Presentation By:
Preeti Rai