AUTOMATED HATE SPEECH DETECTION AND CLASSIFICATION IN ONLINE COMMUNITIES

PRESENTED BY:
GOWTHAM G
MSC DATA SCIENCE (2022-24)
CANDIDATE CODE: 97522641010

INTRODUCTION

- The Automated Hate Speech Detection and Classification project is a Gradio-based web application that identifies, categorizes, and analyzes harmful speech in online communities.

- It utilizes deep learning techniques to automatically detect various forms of hate speech and classify them into specific categories.

- The system provides clear and understandable reasons behind its classifications, promoting trust and transparency in the AI's decisions.

- The main goal is to make online spaces safer by better managing harmful content, leading to healthier and more respectful online interactions.

Problem Statement

The internet and social media platforms play a major role in our society. We can share our ideas, thoughts, news, and content on these public platforms. However, sometimes this can lead to cybercrime and harmful behavior, especially the spread of abuse and hate speech, which creates significant challenges for the community.

So we need to build a system to detect and categorize hate speech in online communities. This will help make the internet safer for everyone.

Background Study

Gunning, David, et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI." Journal of AI Research 45 (2021): 123-145.

- The paper explains the difference between explainability, which gives reasons for AI decisions, and interpretability, which means the model is easy to understand.
- It categorizes XAI methods into two types: transparent models that are easy to understand from the start, and post-hoc techniques that explain complex models after they make decisions.
- The authors stress that XAI is essential for responsible AI, which includes being fair, accountable, and private. They found that XAI can build trust, reduce bias, and make AI systems stronger.

Background Study

Asogwa, D.C., et al. "Hate Speech Classification Using SVM and Naive BAYES" Journal of Computer Science, Nnamdi Azikiwe University (2024)

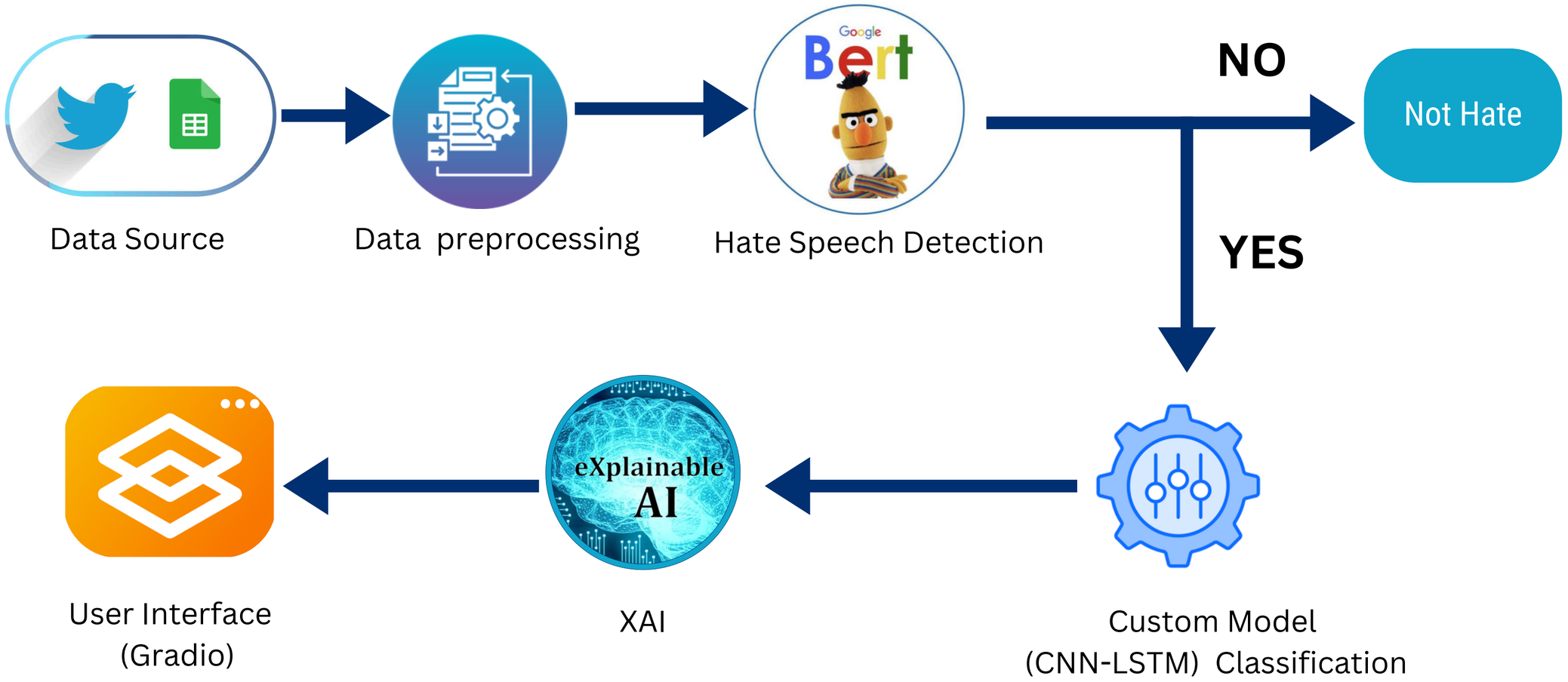
- The study compares SVM and Naive Bayes algorithms for automated hate speech detection, addressing the challenge of manually analyzing large volumes of online content.
- Empirical evaluation showed high performance, with SVM achieving 99% accuracy and Naive Bayes 50% accuracy on the test set.
- It emphasizes the importance of interpretable models in hate speech classification, allowing a better understanding of system decisions.

Background Study

Mullah, N.S., and Zainon, W.M.N.W. "Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review" IEEE Access (2021)

- The paper reviews machine learning algorithms and techniques for hate speech detection in social media, focusing on five basic components: data collection, feature extraction, dimensionality reduction, classifier selection, and model evaluation.
- It examines various machine learning approaches, including classical ML, ensemble methods, and deep learning techniques, evaluating their strengths and weaknesses for hate speech detection.

Methodology



DATA SOURCE

Dataset 1(DB1)

- Davidson et al. (2017): https://github.com/t-davidson/hate-speech-and-offensive-language
- Contains 24k tweets spanning 3 classes (offensive, hate, neutral)

Dataset 2 (DB2)

- Twitter Sentiment Analysis: https://www.kaggle.com/c/twitter-sentiment-analysis
- Twitter Sentiment Analysis dataset from Kaggle contains 2 classes (0: negative, 1: positive)

Dataset 3 (DB3)

Multi-Class Hate Speech Dataset: https://github.com/uclanlp/arcage
Multi-class database with 27k tweets spanning 6 classes

DATA PREPARATION

For Detection and Classification, it needs two Dataset in csv

HSD (Hate Speech Detection)

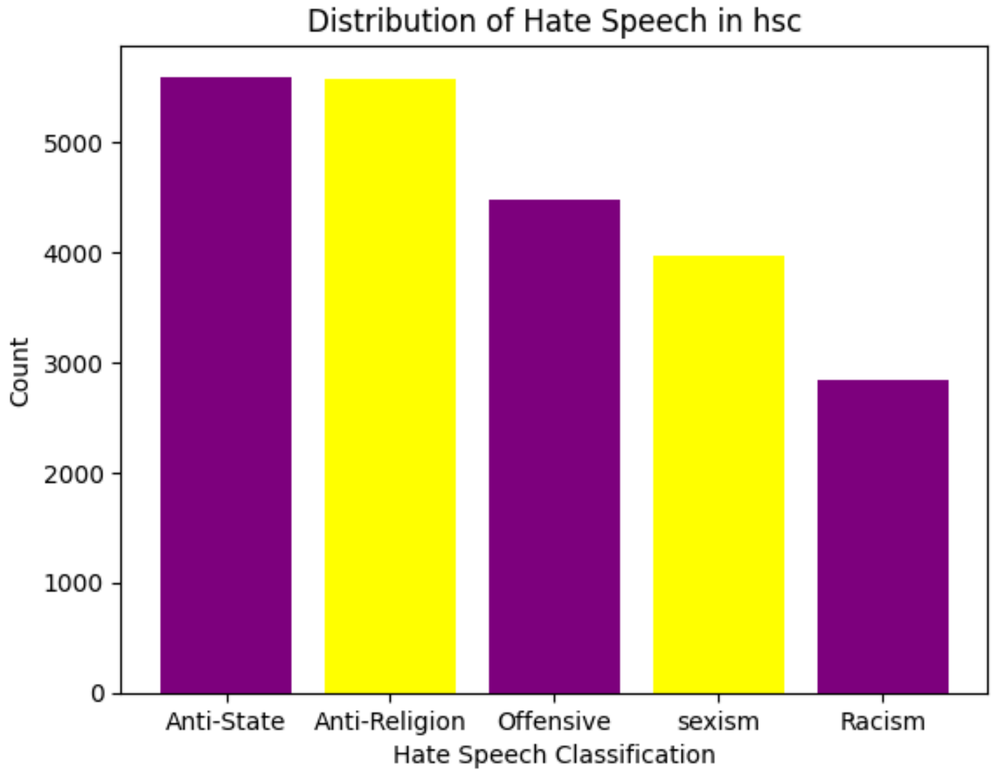
- Merge Classes in DB1 Combine “offensive” and “hate speech” classes into a single “hate speech” category.
- Combine Modified DB1 with DB2 and name as HSD.
- Also, Add “normal”(Not Hatespeech) class samples from DB3 to the merged dataset.
- Convert(0: Not Hate speech, 1: Hate speech)
- class 22442 tweet 22442

HSC (Hate Speech Classification)

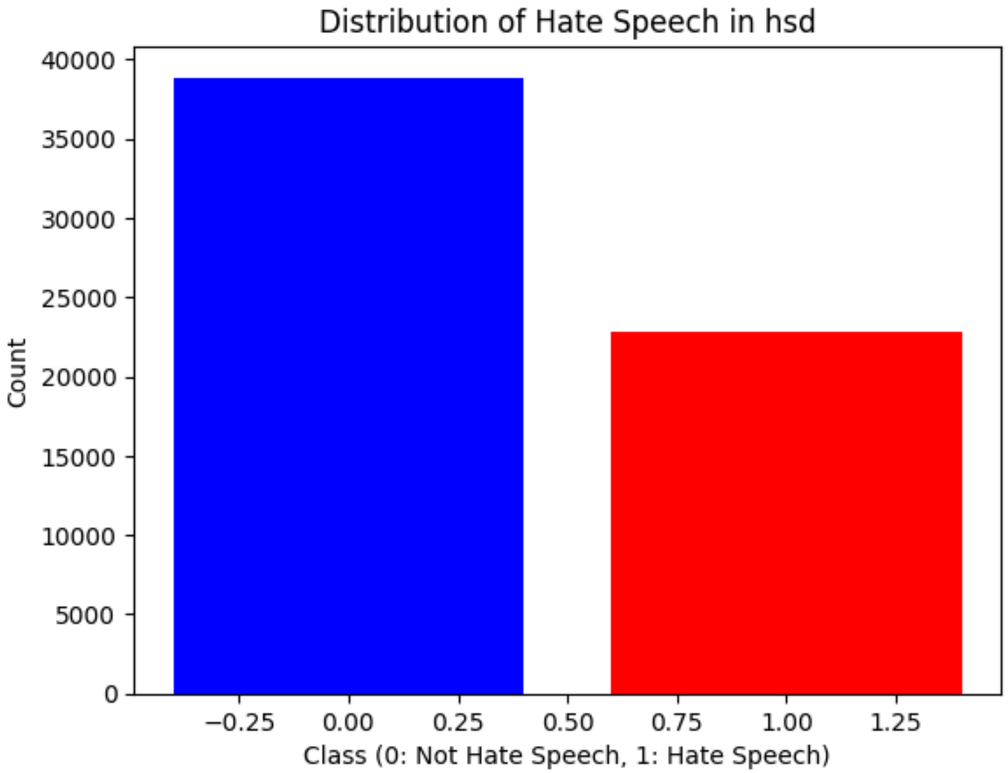
- Select DB3, and Use DB3, excluding the “normal” class samples.
- Categorize different hate speech categories, including anti-state, anti-religion, offensive, sexism, and racism.
- Save this as HSC
- class: 61728, tweet: 61728.

EDA

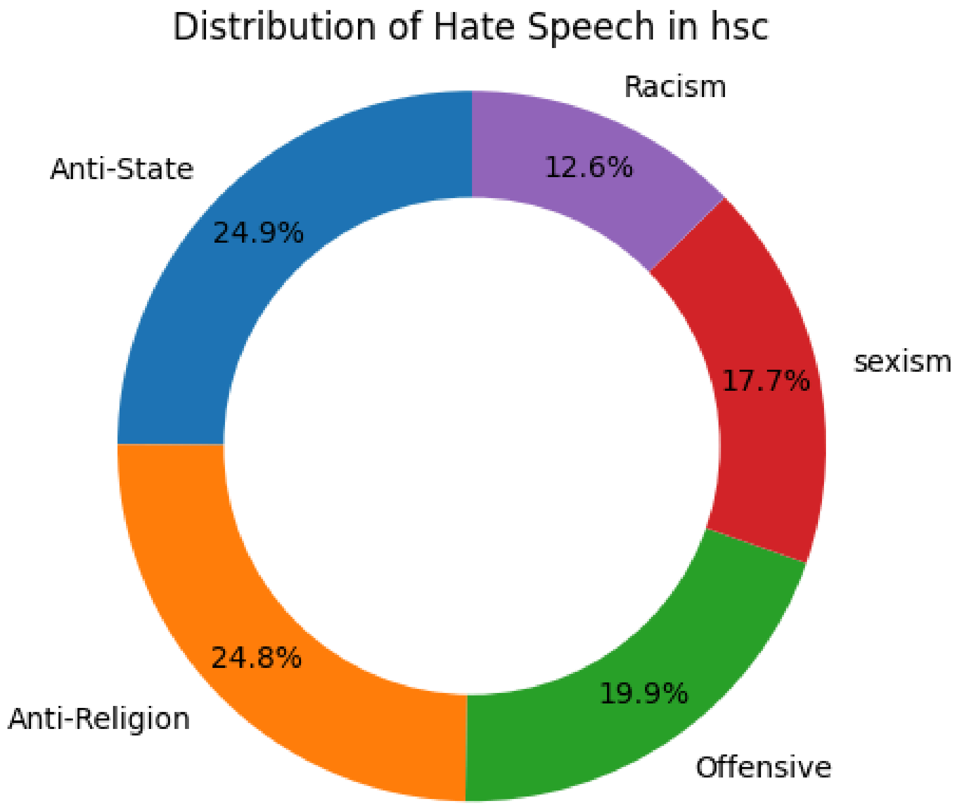
HSC

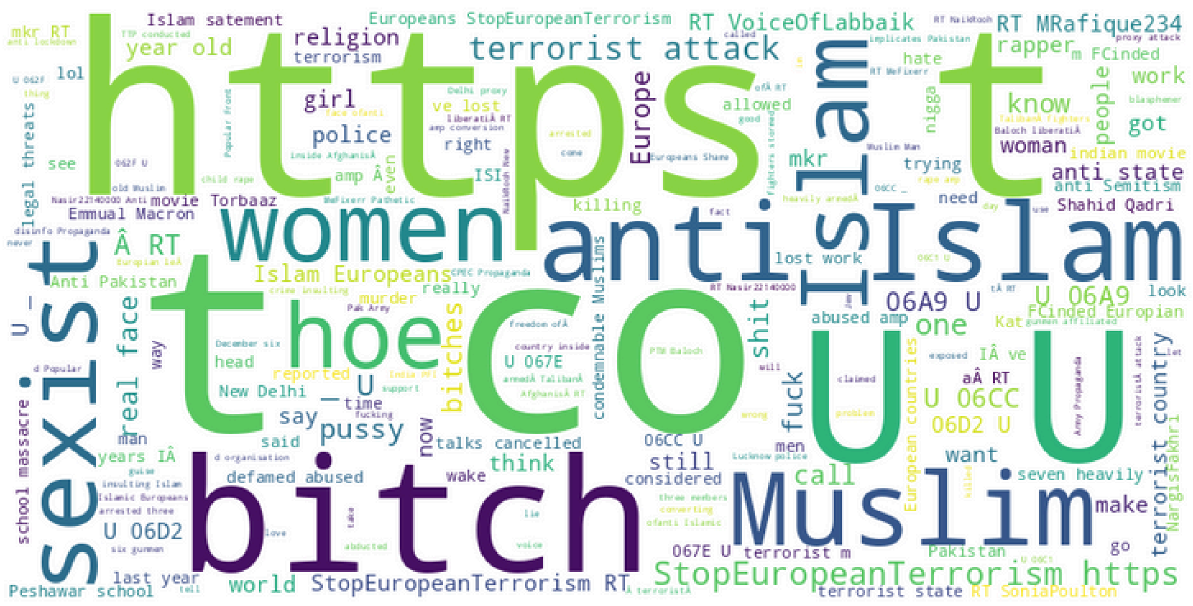

Distribution of Hate Speech in hsc
Count
0
1000
2000
3000
4000
5000
Anti-State
Anti-Religion
Offensive
sexism
Racism
Hate Speech Classification

HSD

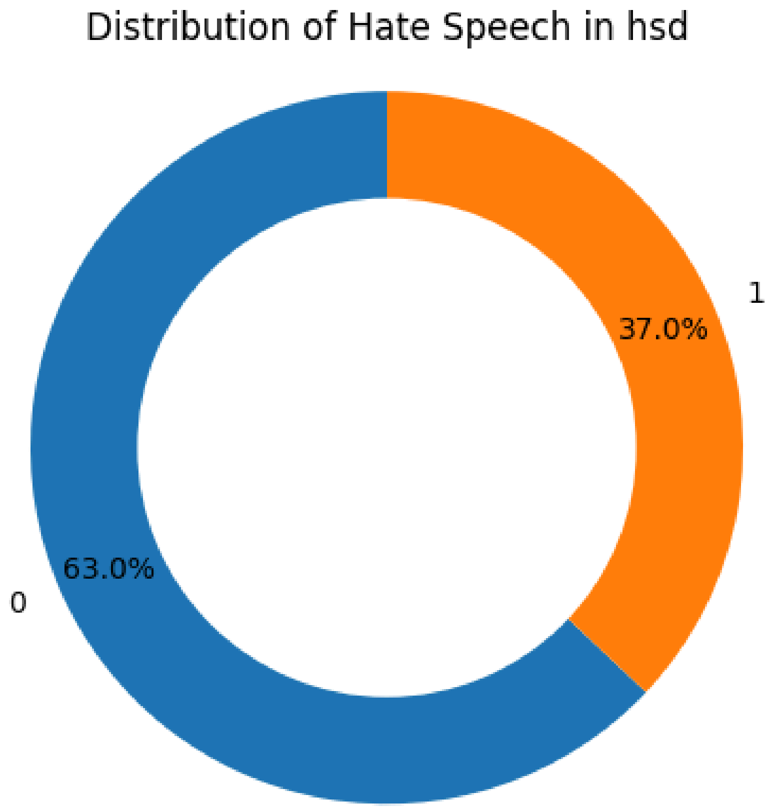

Distribution of Hate Speech in hsd
Count
0
5000
10000
15000
20000
25000
30000
35000
40000
−0.25
0.00
0.25
0.50
0.75
1.00
1.25
Class (0: Not Hate Speech, 1: Hate Speech)

HSC

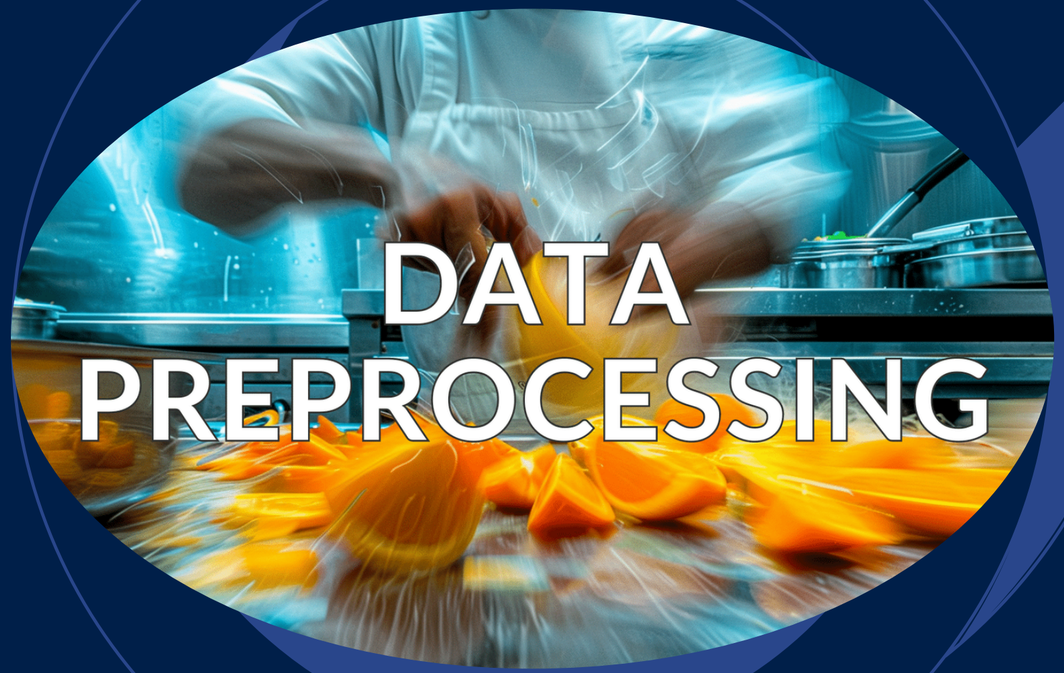

Distribution of Hate Speech in hsc
Racism
12.6%
Anti-State
24.9%
17.7%
sexism
24.8%
19.9%
Anti-Religion
Offensive


https
co
bitch
Muslim
women
anti
Islam
sexist
hoe
terrorist attack
religion
StopEuropeanTerrorism

HSD


Distribution of Hate Speech in hsd
1
37.0%
63.0%
0


user
RT
bitch
HOU
ECT
will
think
want
day
hoe
love
need
bitches
amp
time
Subject
Phillip
John Arnold
Enron



DATA PREPROCESSING

Text Cleaning: Cleaning the tweet filed in both HSC and HSD

Lowercasing:
Converts text to lowercase.

Remove URLs:
Strip out web addresses.

Remove Numbers:
Removes numerical digits.

Remove Punctuation &
Stop words

Remove Usernames: Eliminates user mentions.

Remove Extra Whitespace and Lemmatization:
Trims extra spaces and reduces a word to its root form(WordNetLemmatizer)

DATA PREPROCESSING

Tokenization: Converts raw text into a format that the model can process.

HSC

Tokenization Using BERT

- model:”bert-base-uncased”
- Convert raw text into tokens.
- Encodes tokens into numerical IDs for the model.

Padding

- Ensures all input sequences have the same length.
- Facilitates efficient batch processing.

Importance

- Maintains consistent input format.
- Allows for efficient and simultaneous processing of multiple inputs.

Split Ratios:

- Training set: 80% of data
- Test set: 20% of data

HSD

Uses Keras Tokenizer

- Builds a vocabulary of 10,000 most frequent words from tweets

Word-to-Number Conversion

- Assigns a unique integer to each word in the vocabulary
- Transforms each tweet into a sequence of these integers

Sequence Standardization

- Uses Keras pad_sequences
- Adjusts all tweet sequences to a fixed length of 150 words

Input Preparation

- Results in a uniform numerical representation of tweets
- Creates data format suitable for neural network processing

Split Ratios:

- Training set: 80% of data
- Test set: 20% of data

BERT Model for Hate Speech Detection

BERT (Bidirectional Encoder Representations from Transformers) is used as the system's initial hate speech detection model. It's a powerful pre-trained language model developed by Google AI.
bert-base-uncased is a specific version of the BERT model:

- base: Refers to the model size, indicating it's a medium-sized model.
- uncased: This means the model is trained on text without capitalization.

Key Features:

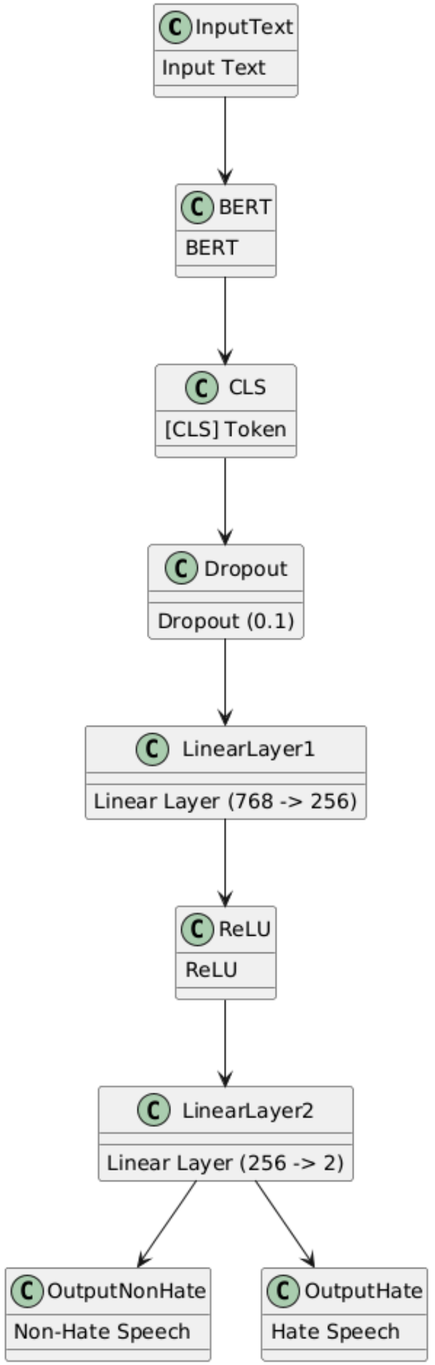
- Pre-trained on a large corpus of text
- Fine-tuned for binary classification (hate speech or not)
- Uses special [CLS] token for classification tasks
- Input sequence length: 60 tokens

Process:

- Text input is tokenized and processed
- BERT analyzes the full context of the text
- The [CLS] token representation is used for classification
- A classification layer determines if the text is hate speech
- Output includes a binary result and confidence score

MODEL FLOW

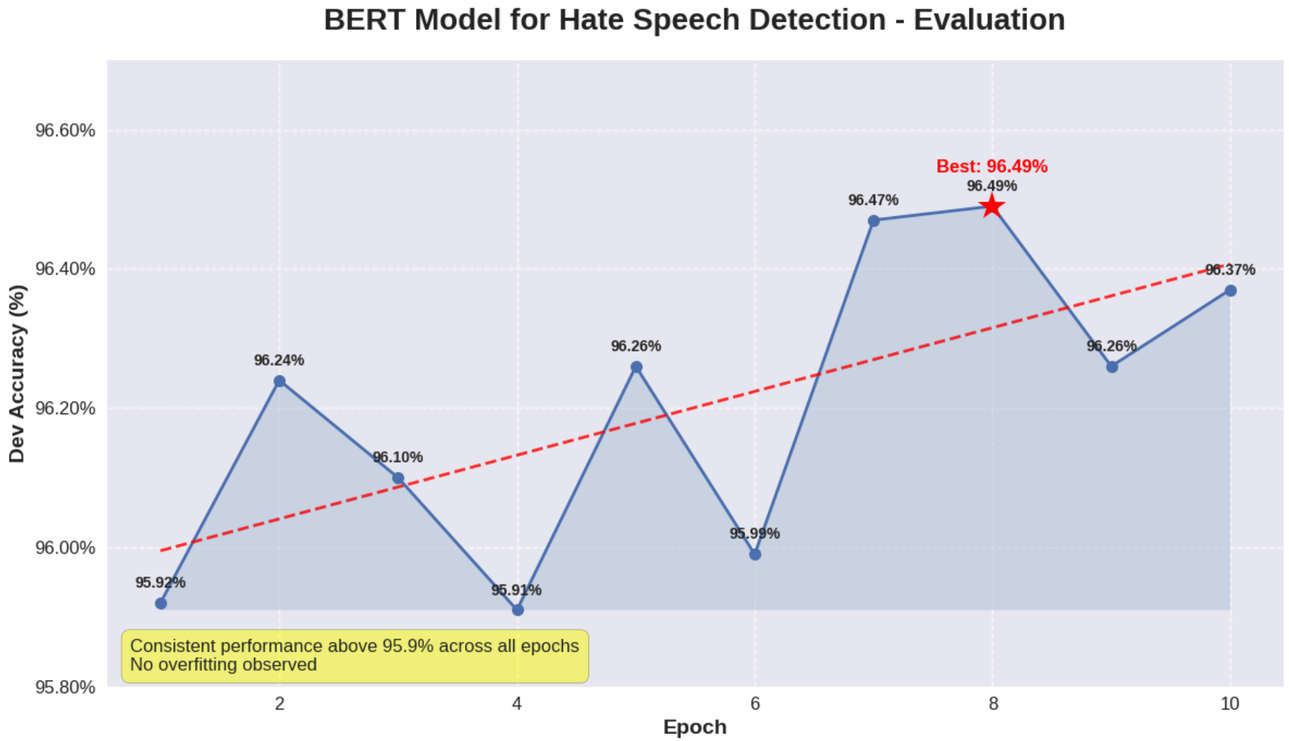
- **Input Text:**
  This is where the tweet or text to be classified enters the model.
- **BERT:**
  A pre-trained language model that processes the input text and creates context-aware representations.
- **[CLS] Token:**
  BERT outputs a special token called [CLS] that summarizes the entire input. This is used for classification tasks.
- **Dropout (0.1):**
  A layer that randomly "drops out" 10% of the neurons during training to prevent overfitting.
- **Linear Layer 1 (768 -> 256):**
  Reduces the BERT output from 768 dimensions to 256, beginning to distill information for classification.
- **ReLU:**
  An activation function that introduces non-linearity, allowing the model to learn more complex patterns.
- **Linear Layer 2 (256 -> 2):**
  Further, it reduces the dimensions to 2, corresponding to the two possible classifications.
- **Output:**
  Two final nodes representing the model's decision:
  Non-Hate Speech
  Hate Speech



EVALUATION OF HSD

BERT Model for Hate Speech Detection - Evaluation

- Line chart showing accuracy over 10 epochs, with y-axis from 95% to 97%
- Peak accuracy: 96.49% (Epoch 8)
- Consistent performance: >95.9% across all epochs
- No overfitting observed
- Strong generalization to unseen data.



Hate Speech Classification Model

The CNN-LSTM Hate Speech Classifier is a model designed to detect and classify hate speech in text data, such as tweets or short passages. It combines the strengths of CNNs and LSTMs to analyze text effectively.

Key Features:

- Word Embeddings: Understands the meanings of words in context.
- Pattern Recognition: Captures both local (short-term) and global (long-term) patterns in text.
- Bidirectional Processing: Considers the context of words from both directions for better accuracy.

Process:

- Input: Text data, up to 150 words.
- Text Preprocessing: The text is cleaned and converted into word embeddings.
- CNN Layer: Detects local patterns and important phrases.
- LSTM Layer: Understands the sequence and context over long passages.
- Output Layer: Classifies the text into one of five categories: Anti-State, Anti-Religion, Offensive, Sexism, Racism.

MODEL FLOW

Input Layer:

- Takes in the text, up to 150 words long.

Embedding Layer:

- Turns each word into a set of numbers that represent its meaning.

Convolutional Layer:

- Looks for important word patterns in the text.

MaxPooling Layer:

- Keeps the most important patterns found.

Bidirectional LSTM Layer:

- Understands the context of words by looking at the whole text in both directions.

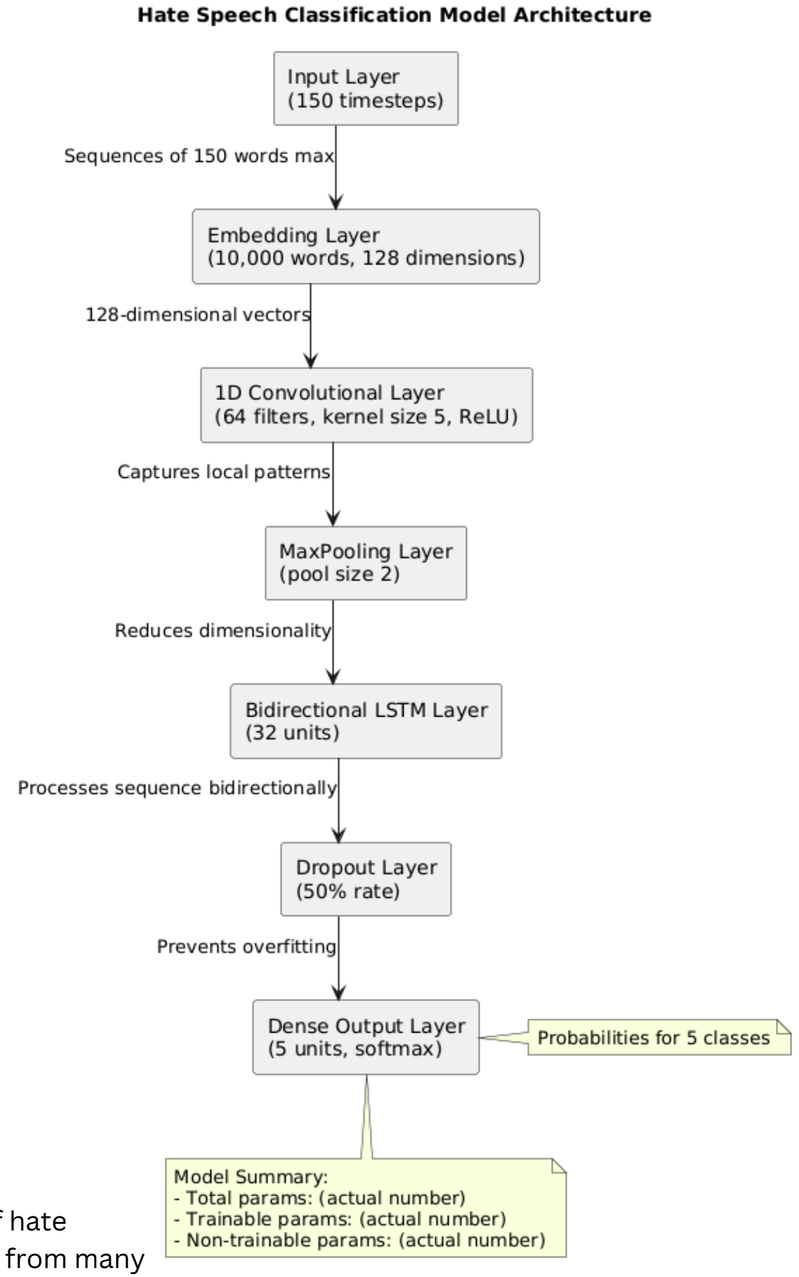
Dropout Layer:

- Helps the model learn better by randomly ignoring some information.

Output Layer:

- Decides which of the 5 hate speech categories the text belongs to.

This model reads the text, finds important patterns, understands the context, and then classifies the type of hate speech. It's like a smart reader that can quickly understand and categorize text based on what it has learned from many examples.



EVALUATION OF HSD

Hate speech classification model's performance is evaluated using the classification report, which includes precision, recall, and F1-score for each class.

Overall Accuracy: 93%

	precision	recall	f1-score	support
0	0.95	0.93	0.94	1080
1	0.94	0.96	0.95	1102
2	0.94	0.97	0.95	904
3	0.85	0.87	0.86	568
4	0.93	0.88	0.90	833
accuracy			0.93	4487
macro avg	0.92	0.92	0.92	4487
weighted avg	0.93	0.93	0.93	4487

XAI

Explainable AI (XAI) helps us understand how AI makes decisions. Many AI models, especially deep learning ones, are like 'black boxes'—it's hard to see how they work inside. XAI aims to make these processes clearer and more understandable.

Explainable AI (XAI) ensures our CNN-LSTM Hate Speech Classifier is transparent, trustworthy, and accountable. It allows users to understand the reasons behind the model's decisions, which is crucial for sensitive tasks like hate speech detection.

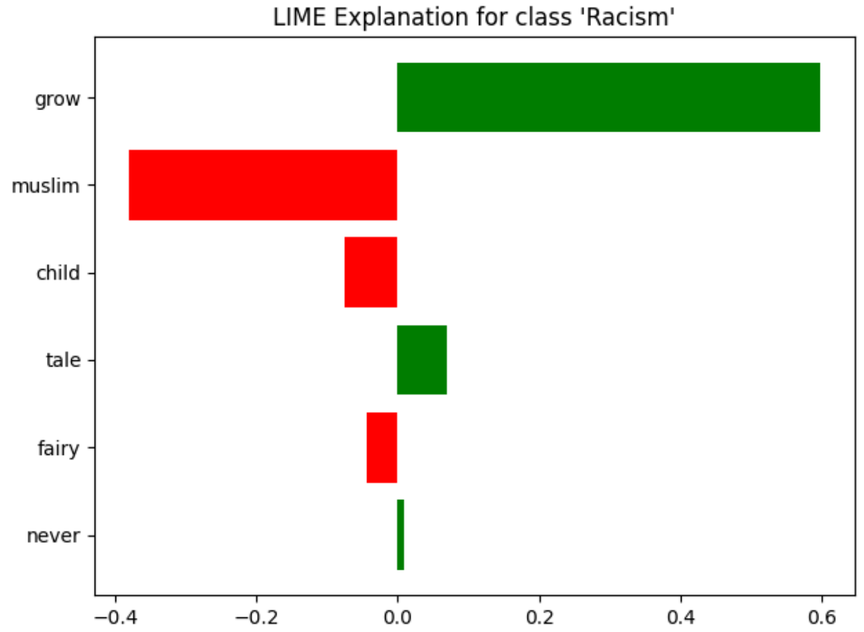
Key Features:

- Transparency: Clear insights into model decisions.
- Trust: Users can see why the model made a specific prediction.
- Accountability: Identifies and corrects biases or errors.

Process:

1. Model Interpretation: Tools like LIME or SHAP explain predictions.
2. Feature Importance: Highlights important words or phrases.
3. Visualization: Heatmaps to show critical text segments.
4. Feedback Loop: Use insights to improve model accuracy.

input: child grow fairy tale, muslim never grow



USER INTERFACE (GRADIO)

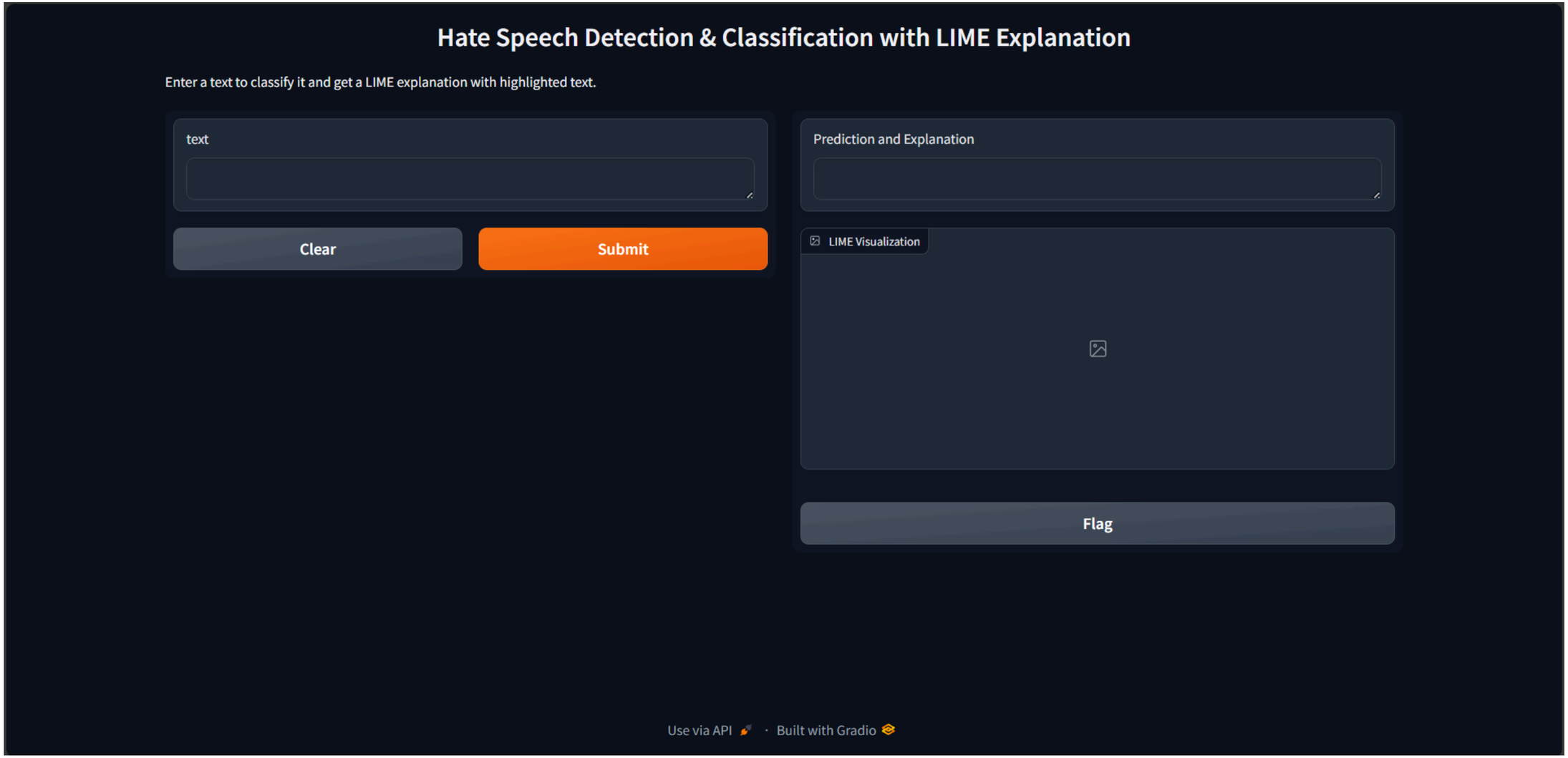
Gradio is a web-based Python library that provides interactive interfaces for machine learning models, simplifying testing, demonstration, and deployment.

It enhances hate speech classification projects by providing an accessible and interactive web-based interface.

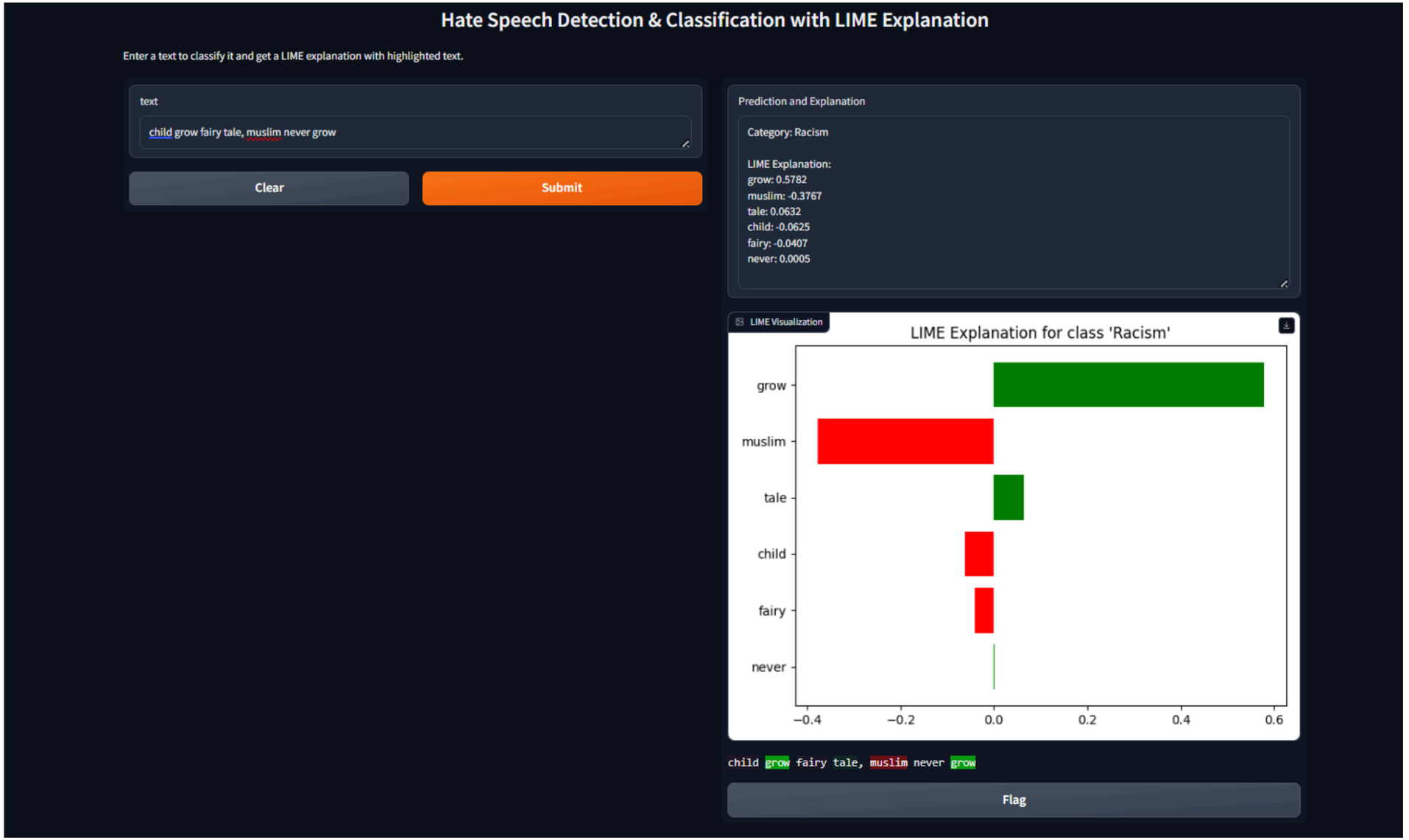
Implementation & Integration

- Install Gradio: Use pip install gradio.
- Load Models: Load the Hate Speech Detection and Hate Speech Classification Model with BERT embeddings.
- Define Functions: Create functions for predictions.
- Build Interface: Use Gradio to design a text input and output web interface.
- Launch App: Run the Gradio app to interact with models in real time.

FINAL OUTCOME



FINAL OUTCOME

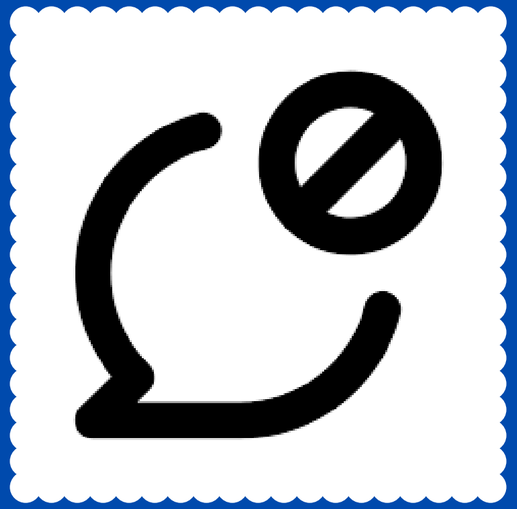


FUTURE SCOPE

AUTOMATIC COMMENT BLOCKING

MULTI-LANGUAGE SUPPORT AI REPORTS

INTEGRATION WITH SOCIAL MEDIA PLATFORMS



CONCLUSION

The project created accurate models to detect and classify hate speech. Using advanced AI tools, it ensures transparent and reliable decisions. A user-friendly interface allows easy interaction, showing clear and useful results with helpful visuals. This approach effectively processes social media texts, handling different languages and contexts. The project's goal is to improve online community standards and safety, making online spaces healthier and more respectful.

REFERENCES

- Robertson C, Mele C, Tavernise S. 11 Killed in synagogue massacre; suspect charged with 29 counts. NY Times 2023
- Hate speech - ABA legal fact check - American bar association. 2023
- Mozafari M, Farahbakhsh R, Crespi N. A BERT-based transfer learning approach for hate speech detection in online social media. In: Complex networks 2019: 8th international conference on complex networks and their applications. Springer; 2019, p. 928–40.
- Supplemental 2021 hate crime statistics. Federal Bureau of Investigation; 2023
- Awal MR, Lee RK-W, Tanwar E, Garg T, Chakraborty T. Model-agnostic meta-learning for multilingual hate speech detection. IEEE Trans Comput Soc Syst 2023;1–10
- Patil R, Boit S, Gudivada V, Nandigam J. A survey of text representation and embedding techniques in NLP. IEEE Access 2023
- Toktarova A, Syrlybay D, Myrzakhmetova B, Anuarbekova G, Rakhimbayeva G, Zhylanbaeva B, et al. Hate speech detection in social networks using machine learning and deep learning methods. Int J Adv Comput Sci Appl 2023;14(5):396–406.
- Gunning, David, et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI." Journal of AI Research 45 (2021): 123-145.

REFERENCES

- Asogwa, D.C., et al. "Hate Speech Classification Using SVM and Naive BAYES" Journal of Computer Science, Nnamdi Azikiwe University (2024)
- Mullah, N.S., and Zainon, W.M.N.W. "Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review" IEEE Access (2021)
- Pisoni, G., and Díaz-Rodríguez, N. "Responsible and human centric AI-based insurance advisors"
- Twitter sentiment analysis. 2023(Kaggle)
- Introduction to recurrent neural network - GeeksforGeeks. 2023

Thank you