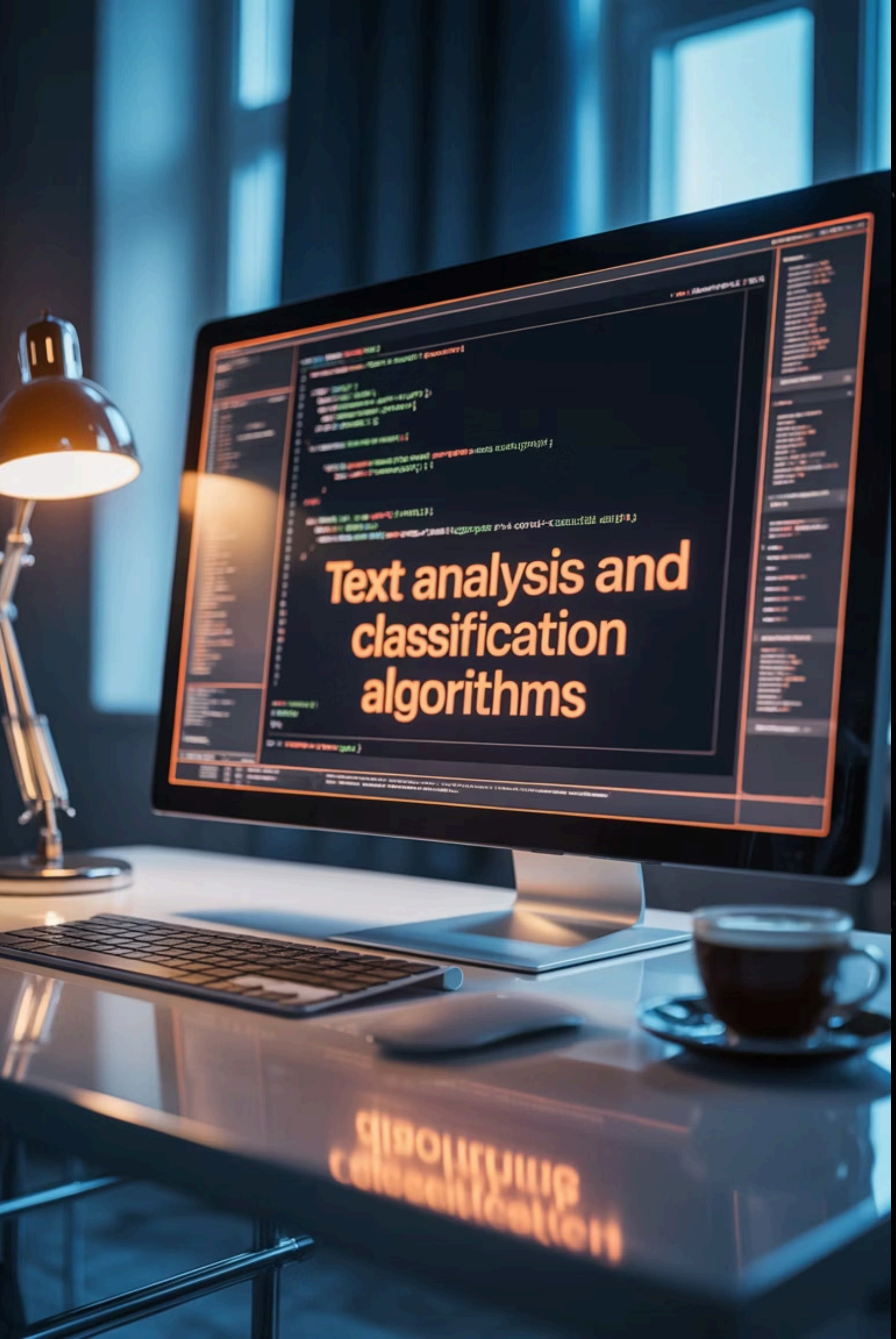


LLM Chatbot with Content Filtering and Hate Speech Detection Using Machine Learning

Dataset : Online Hate Speech Dataset (Kaggle)

Member 1	Member 2	Member 3
Shreya Lal	Vedant Shrikhande	Sonalika Chauhan
25030242054	25030242056	25030242057

Objective

Detect hate speech in text using ML models.

Dataset: Dynamically generated hate speech dataset.

Main Goal: Build & evaluate a text classification pipeline.

Libraries & Tools Used

pandas

Data handling

scikit-learn

Train/test split, ML models, evaluation

TfidfVectorizer

Text feature extraction

KaggleHub

Dataset download

Platform: Google Colab Notebook (Python)

Dataset Overview

- Source: usharengaraju/dynamically-generated-hate-speech-dataset (Kaggle)
- Format: CSV
- Columns: id, text, label
- Labels: Hate speech vs Non-hate speech

Data Exploration

01

Shape of dataset (rows × columns).

02

Checked for missing values.

03

Label distribution (balanced/unbalanced).

04

Sample rows shown for inspection.

Data Cleaning

Retained only relevant columns (id, text, label).

Dropped rows with missing text or label.

Converted text to lowercase.

Verified unique labels.

Train-Test Split

- Used train_test_split (80% train, 20% test).
- Stratified split to maintain label balance.
- Printed sample training data (text + label).

Feature Engineering

Used TF-IDF Vectorizer:

Removed English stopwords.

Limited to 5000 max features.

Converted text into numerical feature vectors.

Output: Sparse matrix (X_train, X_test).

Model Training

- Tried multiple ML algorithms (likely: Logistic Regression, Naive Bayes, SVM, etc. — I'll confirm from later code cells).
- Models trained on TF-IDF features.
- Stored training accuracy results.

Model Evaluation

- Metrics: Accuracy, Precision, Recall, F1-score.
- Compared results of different models.
- Best-performing model identified.

Main Output

- Classification performance summary.
- Demonstrated predictions on unseen test data.
- Example: Input text → Model prediction (Hate/Non-hate).

Key Insights

Text preprocessing & feature extraction are crucial.

Model performance depends on balancing dataset.

TF-IDF + ML classifier works effectively for hate speech detection.

Limitations

Dataset size limited.

Bias possible in data labeling.

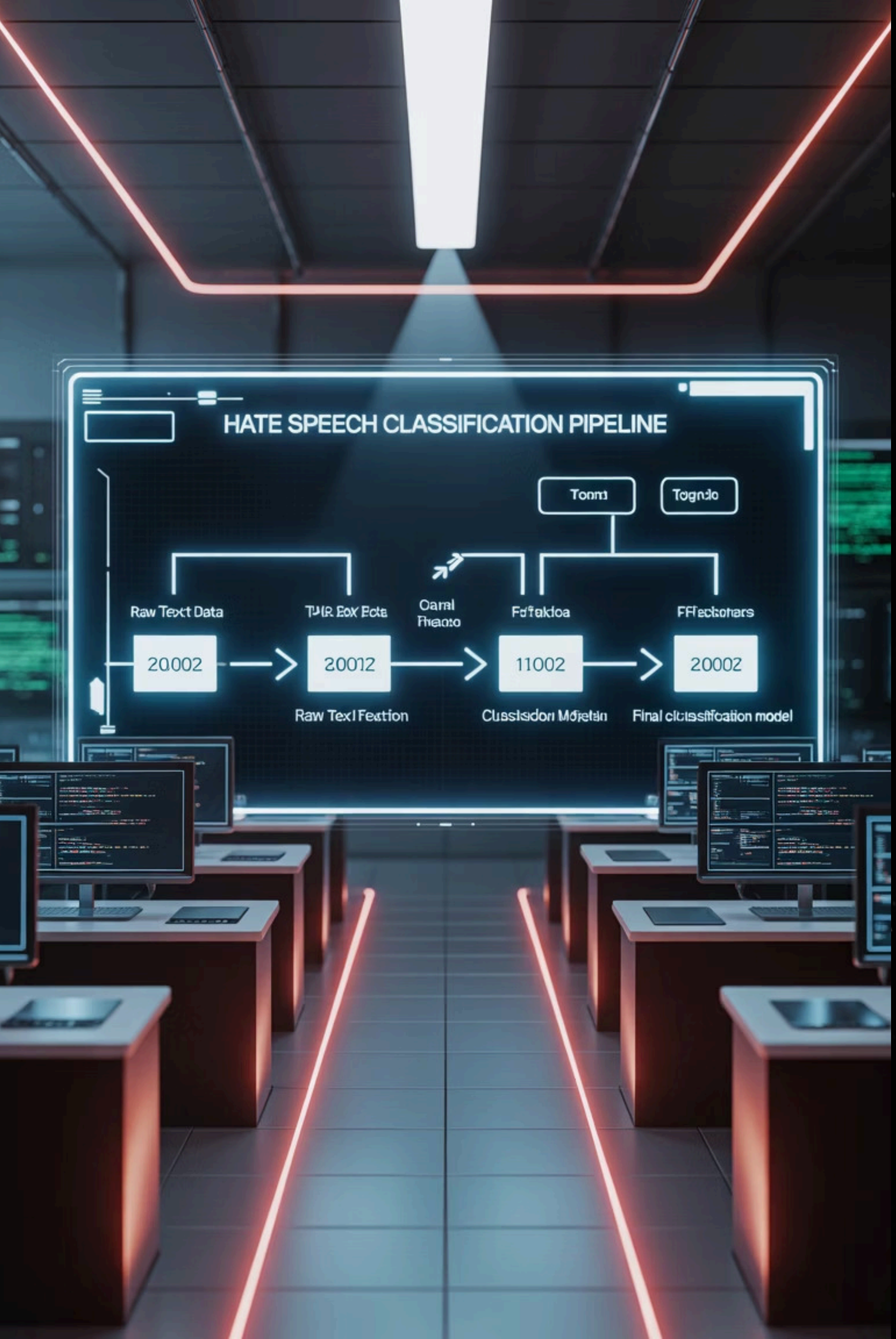
Contextual understanding (sarcasm, implicit hate) may be missed.

Future Work

Explore deep learning (RNN, Transformers).

Use word embeddings (Word2Vec, BERT).

Expand dataset to cover more contexts.



Conclusion

> Successfully built ML pipeline and chatbot for hate speech detection.

End-to-end workflow: Data → Preprocessing → Feature Extraction → Modeling → Evaluation → Chatbot