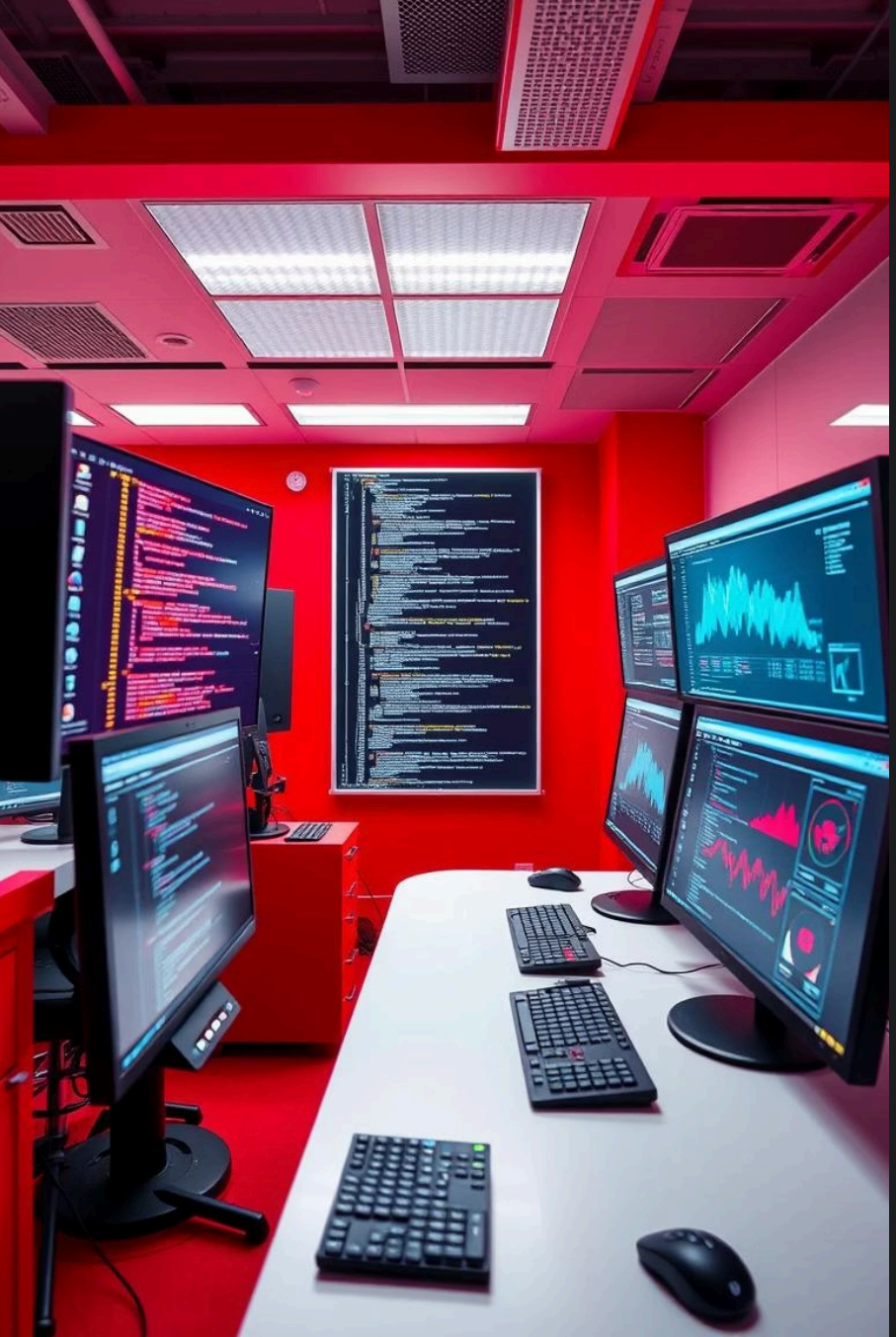


AI vs Human Text Classification: Dataset and Model Insights

Presented By:

Huiting Wu, Sharmeen Kapoorwala, Stella Kent

Dataset Overview and Initial Exploration

Dataset Size

The dataset contains 487,235 text samples labeled as human (0) or AI-generated (1).

Label Distribution

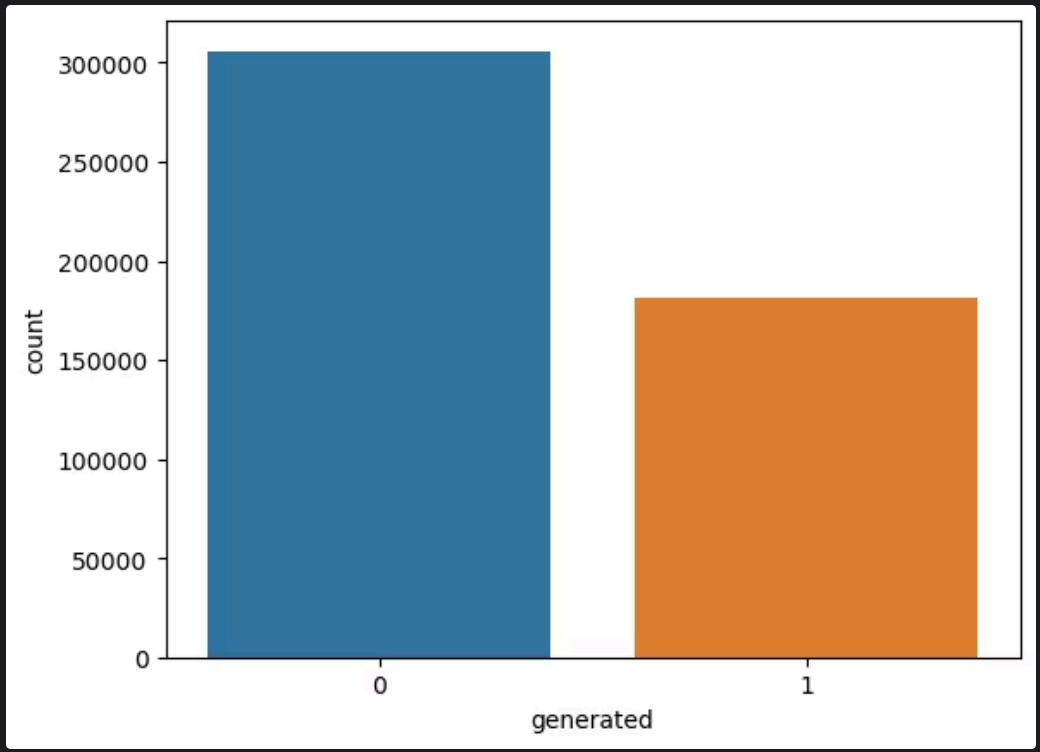
Two classes are present, with labels converted to integers for modeling.

Initial Visualization

A count plot shows the balance between human and AI-generated texts.

Data Visualizations

	text	generated
0	Cars. Cars have been around since they became ...	0.0
1	Transportation is a large necessity in most co...	0.0
2	"America's love affair with it's vehicles seem...	0.0
3	How often do you ride in a car? Do you drive a...	0.0
4	Cars are a wonderful thing. They are perhaps o...	0.0



Objective

Can we build a model and a Word Cloud that will help us predict whether an essay is AI generated or if it is written by a human?

Text Preprocessing Steps

1 Lowercasing

All text converted to lowercase to standardize input.

2 Handling Negations

Negative contractions expanded for clarity in tokenization.

3 Tokenization & Stopword Removal

Text split into tokens, removing common stopwords to reduce noise.

4 Lemmatization

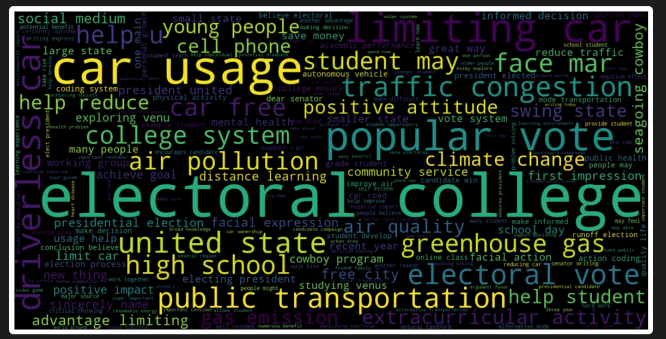
Tokens lemmatized to their base forms for consistency.

Visualizing Text Data with Word Clouds

Human Text Word Cloud

```
from collections import Counter
Counter(' '.join(df[df['generated'] == 0]['text']).split()).most_common(10)
```

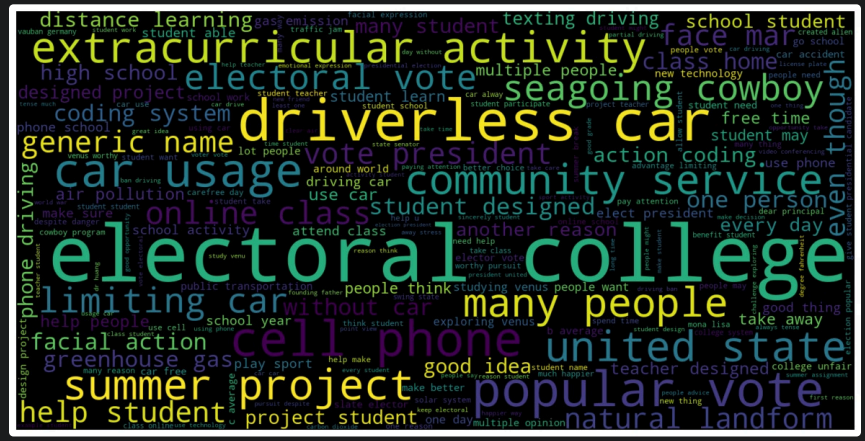
```
[('student', 1168211),
 ('car', 918050),
 ('people', 899415),
 ('would', 885465),
 ('school', 623874),
 ('could', 459994),
 ('get', 459319),
 ('time', 448035),
 ('one', 423813),
 ('like', 421933)]
```



AI-Generated Text Word Cloud

```
from collections import Counter
Counter(' '.join(df[df['generated'] == 1]['text']).split()).most_common(10)
```

```
[('student', 421412),
 ('car', 321614),
 ('people', 273132),
 ('also', 239023),
 ('help', 225183),
 ('electoral', 220390),
 ('like', 207265),
 ('college', 205421),
 ('state', 198531),
 ('may', 191369)]
```



Train-test split

```
# split the data

x_train, x_test, y_train, y_test = train_test_split(df['text'], df['generated'],
                                                    test_size = 0.3,
                                                    stratify = df['generated'], # for imbalance data
                                                    random_state = 42)
```

- train_test_split()
- stratify
- Training Set : 70%
  Testing Set : 30%

Feature Extraction with TF-IDF Vectorization

- fit_transform()
- transform()
- converted into matrix for machine learning models

Model Training and Performance

Logistic Regression

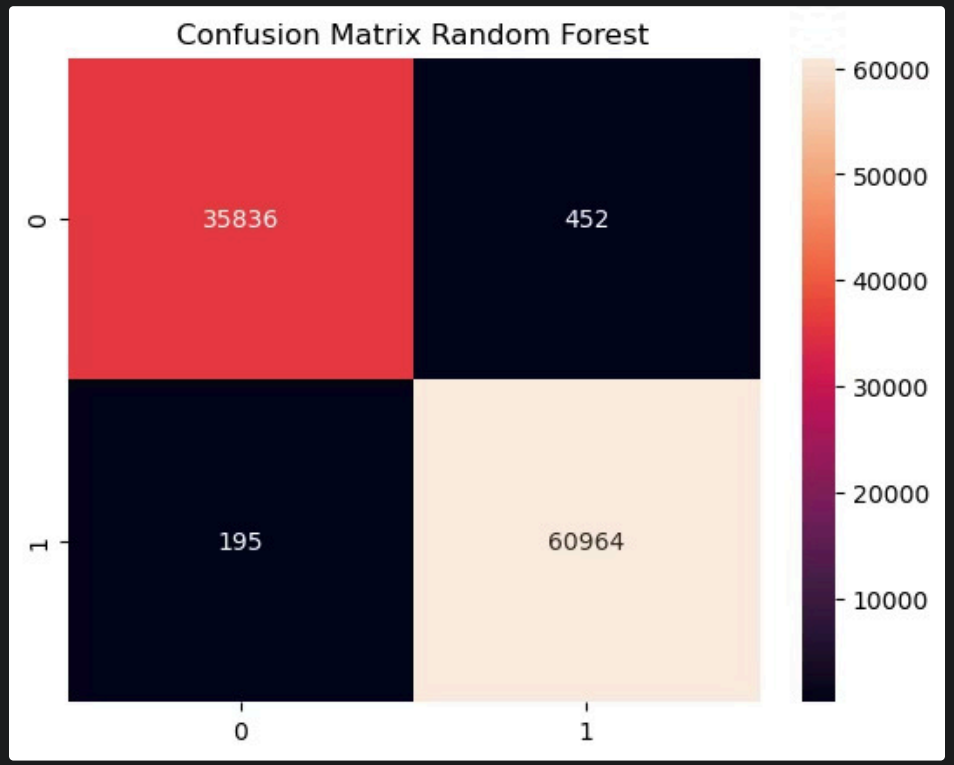
- Trained with 20,000 iterations
- achieved 99.3% accuracy on test data.

XGBoost Classifier

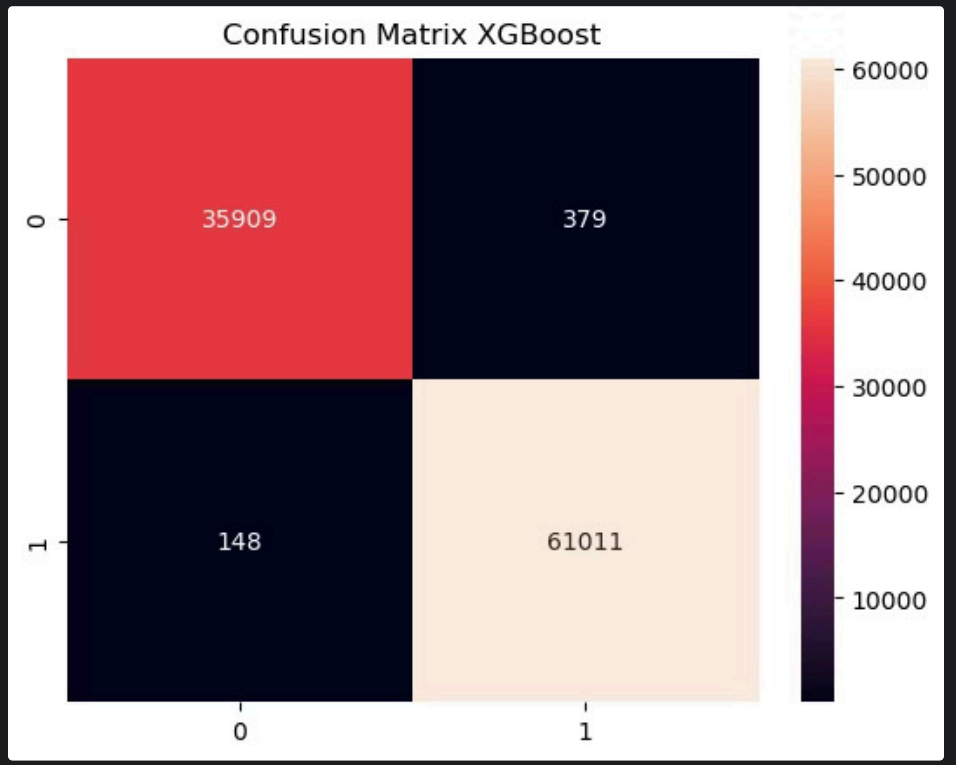
- Achieved slightly higher accuracy of 99.5%
- Demonstrates strong predictive power.

Confusion Matrix Analysis

Logistic Regression Matrix



XGBoost Matrix



Conclusion & Next Steps

Effective Text Classification

XGBoost slightly better than Logistic Regression

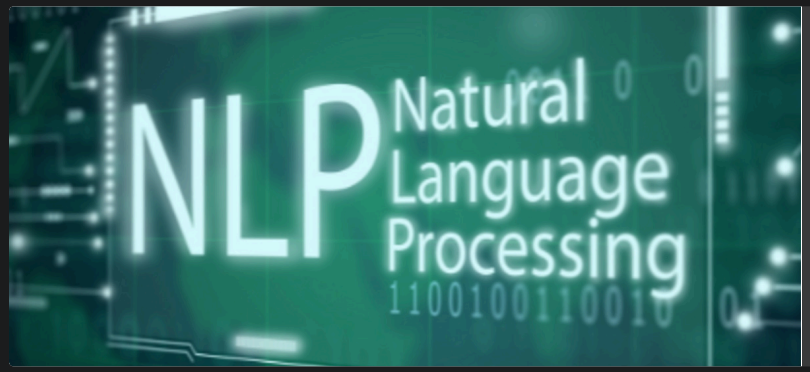
Importance of Preprocessing

Data preprocessing and TF-IDF were effective for text representation.

Future Work

Explore additional models and real-time detection applications to enhance robustness.

Reference



www.kaggle.com

AI Vs Human Text

500K AI and Human Generated Essays

Thank You! 😊