Detecting AI generated text

Amr Kandil, Lukas Donner, Mohamad Hussen Chamsi

Detecting AI vs. Human-Written Text

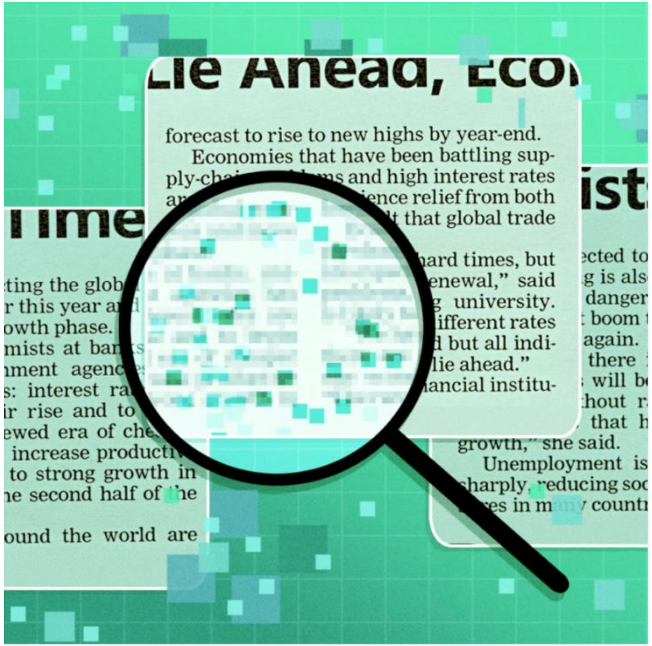
Introduction

- AI and large language models (LLMs) like GPT-3, PaLM, and ChatGPT are rapidly advancing.
- These models can answer complex questions on topics like science, math, and history .
- Newer models can even gather real-time data from the internet, making them highly useful.

Challenges

- It's becoming harder to tell if content was written by a human or AI.

- Problems caused by AI-generated text:
    - Plagiarism
    - Fake information
    - Misinformation



Supported SDGs

SDG 4: Quality Education

- Helps schools and universities detect AI-generated content.
- Ensures academic integrity and fairness in education.

SDG 16: Peace, Justiceand Strong Institutions

- Limits the spread of false information.
- Builds trust and transparency for peaceful societies.

Supported SDGs

SDG 9: Industry, Innovation, and Infrastructure

Our project support responsible AI development by creating systems to manage its risks Our project share to building strong and trust digital infrastructures.

SDG 17: Partnerships for the Goals

Collaborates with educational institutions and tech companies.

Dataset Overview

Kaggele Base Dataset

Dataset Structure:

- Essays written in response to one of seven essay prompts.
- Training set contains essays from two prompts; other five prompts are part of the hidden test set.

The target column indicates whether an essay is:

- **Student-written (0)**
- **AI-generated (1)**

Files Provided:

train_essays.csv: Training data, including essays and metadata.
train_prompts.csv: Instructions and source text for prompts.
test_essays.csv: Contains dummy test data for validation.

Sample:

„Cars. Cars have been around since they became famous in the 1900s, when Henry Ford created and built the first ModelT. Cars have played a major role in our every day lives since then. But now, people are starting to question if limiting car usage would be a good thing. To me, limiting the use of cars might be a good thing to do."

Kaggele Base Dataset

```
train_data.head()
```

	id	prompt_id	text	generated
0	0059830c	0	Cars. Cars have been around since they became ...	0
1	005db917	0	Transportation is a large necessity in most co...	0
2	008f63e3	0	"America's love affair with it's vehicles seem...	0
3	00940276	0	How often do you ride in a car? Do you drive a...	0
4	00c39458	0	Cars are a wonderful thing. They are perhaps o...	0

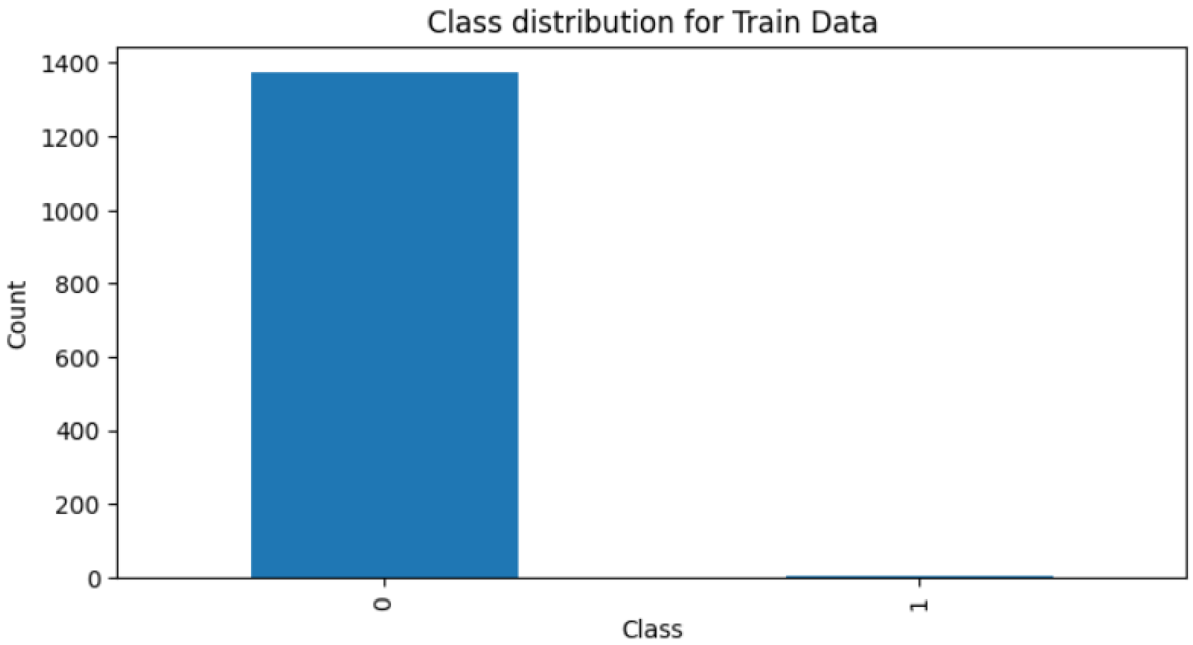
```
train_prompts.head()
```

	prompt_id	prompt_name	instructions	source_text
0	0	Car-free cities	Write an explanatory essay to inform fellow ci...	# In German Suburb, Life Goes On Without Cars ...
1	1	Does the electoral college work?	Write a letter to your state senator in which ...	# What Is the Electoral College? by the Office...

Dataset Imbalance

Initial Imbalance:

- Limited number of AI-generated essays
- Skewed distribution favors student-written essays (class 0)



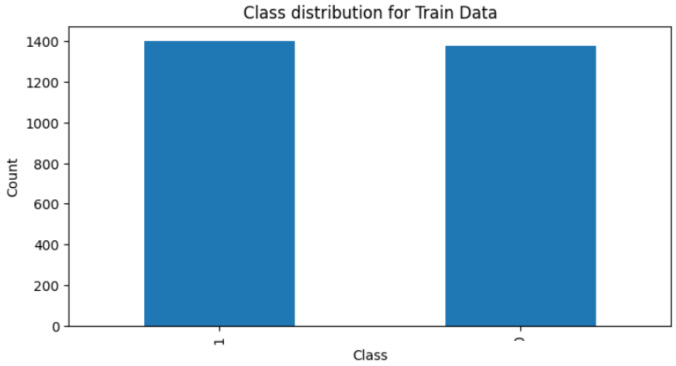
Dataset Imbalance

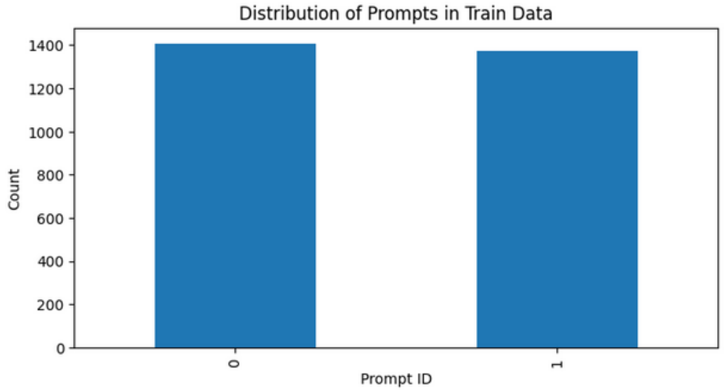
Solution - Data Augmentation:

- Generated additional AI-written essays using OpenAI's GPT-3.5-turbo-0125 model.

- Augmentation replicates the original process:Prompts and source text used as inputs.

- Outputs simulated essays similar to student-written responses.

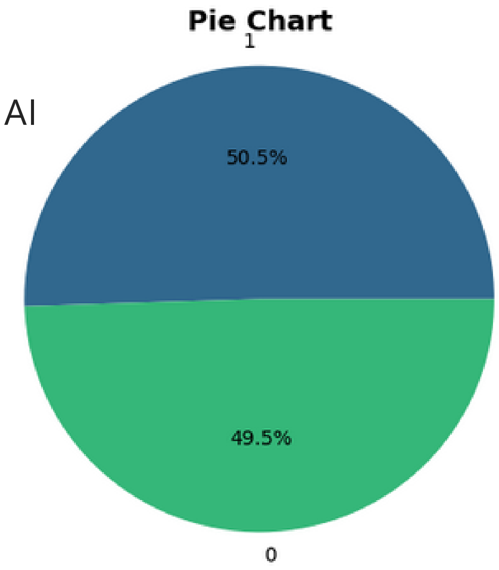


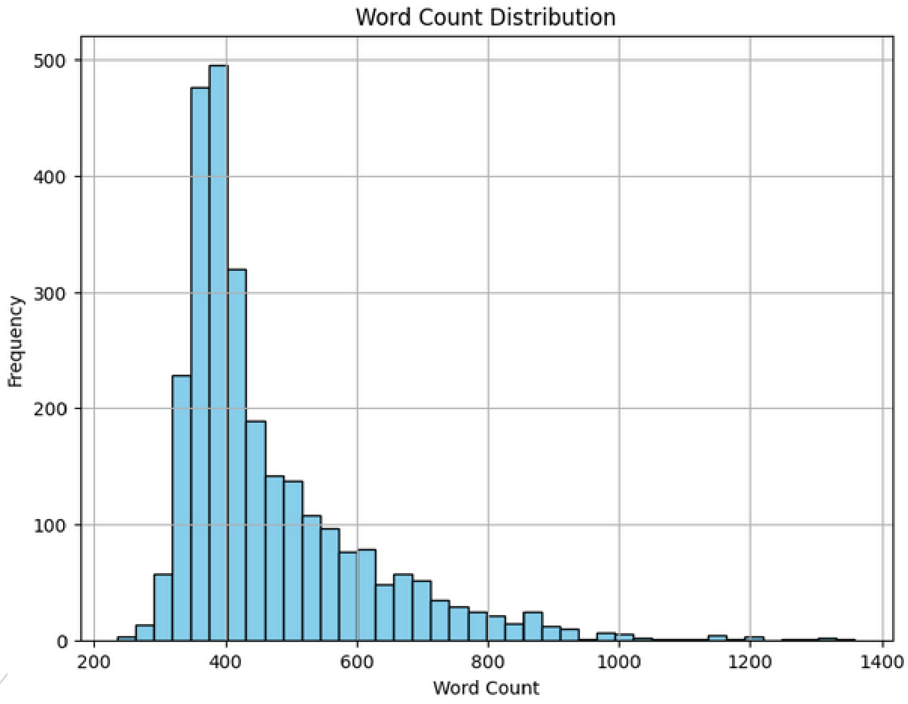


Data visualisation

Distribution by class:

(after augmentation by AI generated texts)





Dataset Preprocessing

Data cleaning 1/2

Data Cleaning is a crucial step in data preprocessing to ensure text data is clean, consistent, and suitable for machine learning models.

Key Steps:

- Removing special characters
- Removing emojis
- Removing URLs
- Retaining periods, question marks, and exclamation marks
- Removing unnecessary white spaces
- Expanding contractions

Result after cleaning:

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

                                        cleaned_text
0  cars. cars have been around since they became ...
1  transportation is a large necessity in most co...
2  americas love affair with its vehicles seems t...
3  how often do you ride in a car? do you drive a...
4  cars are a wonderful thing. they are perhaps o...
```

Data cleaning 2/2

Stop-Words:

•Words like "and," "is," or "the" may be irrelevant for some tasks.

•Their removal depends on the problem context.

Result after cleaning:

```
                                        cleaned_text
0  cars. cars around since became famous s, henry...
1  transportation large necessity countries world...
2  americas love affair vehicles seems cooling sa...
3  often ride car? drive one motor vehicle work? ...
4  cars wonderful thing. perhaps one worlds great...
```

Feature Extraction

Feature Extraction

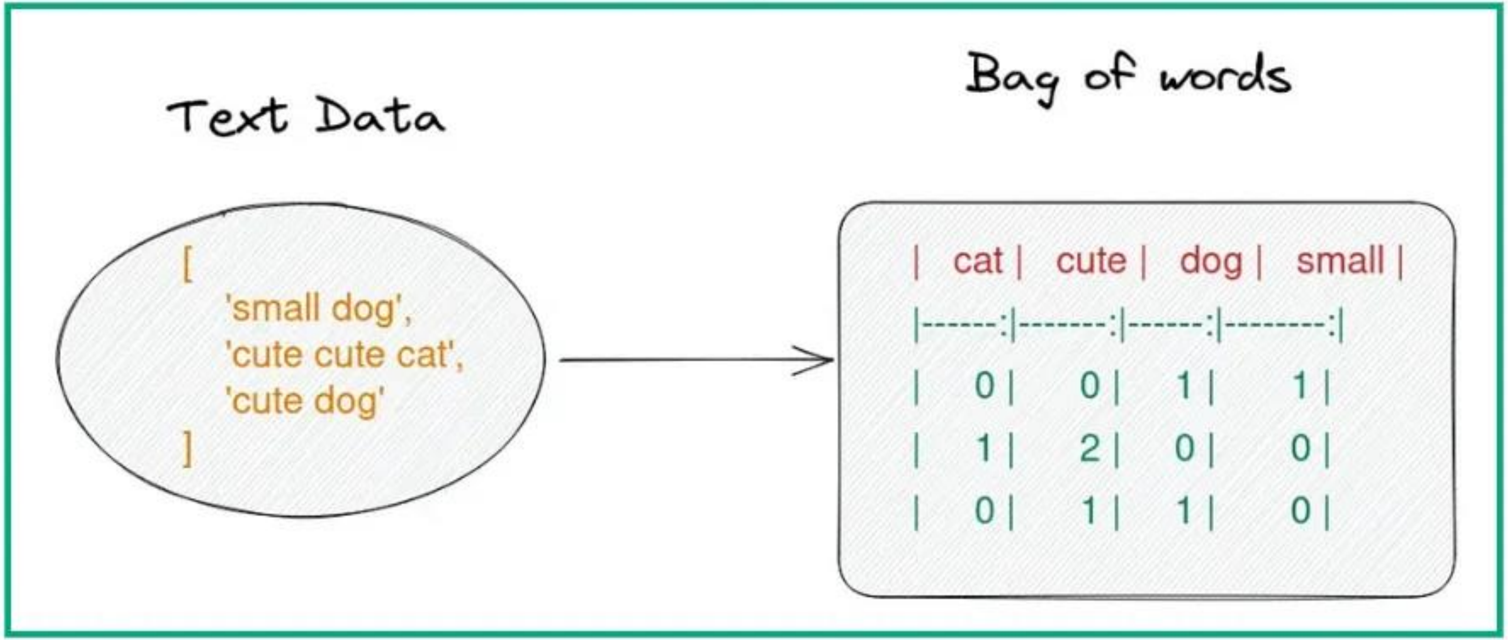
- Text must be converted into numerical format to feed into machine learning models

- These methods capture word importance, relationships, or context, enabling models to make more accurate predictions.

- Popular Feature extraction tools include
- Bag of Words
- TF-IDF
- Embeddings

- In this Notebook we will explore two methods:
- 1. TF-IDF
- 2. Contextual Embeddings with BERT
- - This will be explained and implemented in section 9.2 of the notebook

Bag of Words


Text Data
[
'small dog',
'cute cute cat',
'cute dog'
]
Bag of words
cat	cute	dog	small
0	0	1	1
1	2	0	0
0	1	1	0

TF-IDF

Term Frequency (TF): TF measures the frequency of a term within a document

$$TF(t,d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d}$$

Inverse Document Frequency (IDF): IDF measures the rarity of a term across a collection of documents

$$IDF(t,D) = \log\left(\frac{\text{Total number of documents in the corpus } N}{\text{Number of documents containing term } t}\right)$$

Combining TF and IDF: TF-IDF

TF-IDF is a numerical statistic that reflects the significance of a word within a document relative to a collection of documents

$$\text{TF-IDF}(t,d,D) = TF(t,d) \times IDF(t,D)$$

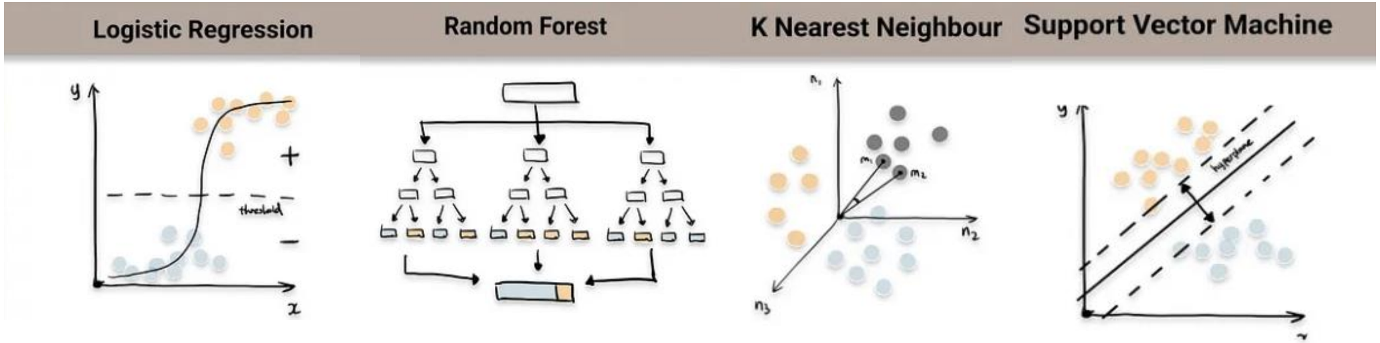
TF-IDF

Index →	0	1	2	3	4	5	6	7	8
	and	**document**	**first**	**is**	**one**	**second**	**the**	**third**	**this**
"This is the first document."	0	0.46979139	0.58028582	0.38408524	0	0	0.38408524	0	0.38408524
"This document is the second document."	0	0.6876236	0	0.28108867	0	0.53864762	0.28108867	0	0.28108867
"And this is the third one."	0.51184851	0	0	0.26710379	0.51184851	0	0.26710379	0.51184851	0.26710379
"Is this the first document?"	0	0.46979139	0.58028582	0.38408524	0	0	0.38408524	0	0.38408524

- The terms are ranked by their overall importance
- This importance score is typically calculated as the sum of TF-IDF values for a term across all documents.
- We pick the top 100 features/words in the Text corpus

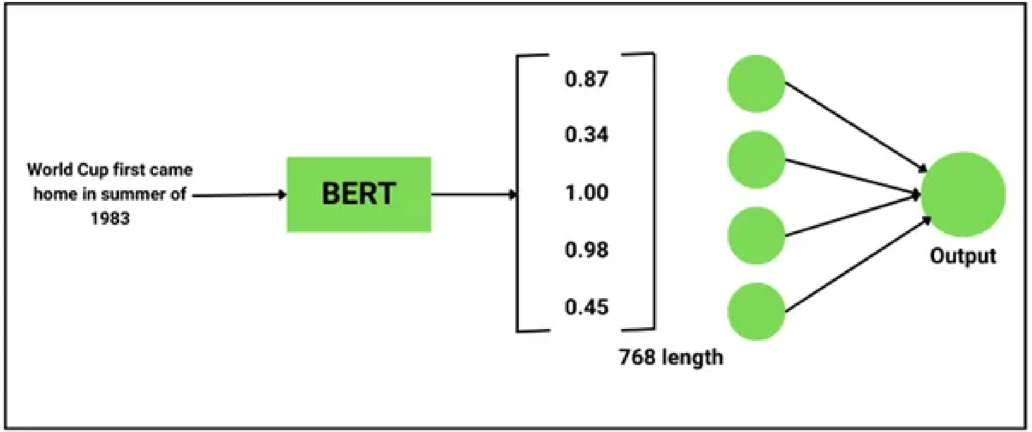
Classification

1. Classical ML (using TFIDF)



Classification

2. Deep NN (using Word Embeddings)



DistilBERTForSequenceClassification

- Transformer-based language model
- Freeze the parameters of Core Distil-BERT
- Only train the classification Head

Phase 1: Analysis

Logistic Regression

Accuracy: 0.9964028776978417

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	413
1	1.00	1.00	1.00	421
accuracy			1.00	834
macro avg	1.00	1.00	1.00	834
weighted avg	1.00	1.00	1.00	834

KNN

Training Accuracy: 99.74%
Test Accuracy: 99.52%
Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	413
1	1.00	1.00	1.00	421
accuracy			1.00	834
macro avg	1.00	1.00	1.00	834
weighted avg	1.00	1.00	1.00	834

Random Forrest

Training Accuracy: 100.00%
Test Accuracy: 99.76%

	precision	recall	f1-score	support
0	1.00	1.00	1.00	413
1	1.00	1.00	1.00	421
accuracy			1.00	834
macro avg	1.00	1.00	1.00	834
weighted avg	1.00	1.00	1.00	834

SVM

SVM Training Accuracy: 99.95%
SVM Test Accuracy: 99.76%

SVM Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	413
1	1.00	1.00	1.00	421
accuracy			1.00	834
macro avg	1.00	1.00	1.00	834
weighted avg	1.00	1.00	1.00	834

Phase 1: Analysis

Distil-BERT

```
Accuracy: 1.00
Precision: 1.00
Recall: 1.00
F1 Score: 1.00
```

- Models are achieving very high accuracies.
- The dataset is relatively small.
- Data leakage is occurring due to similarities between training and test datasets.

Proposals

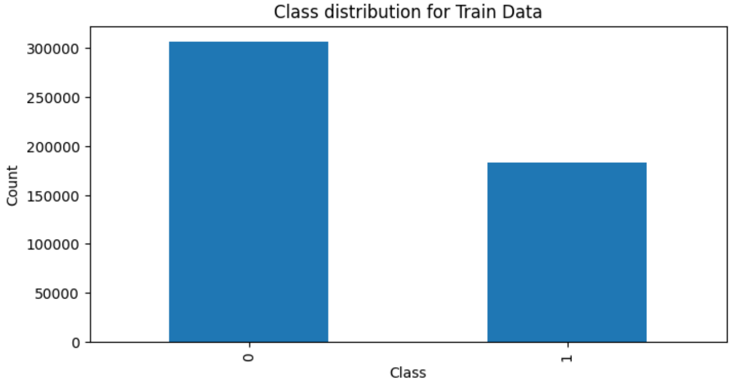
- Increase the dataset size by appending a more varied and larger dataset.
- This approach aims to reduce repetitiveness and mitigate data leakage issues.

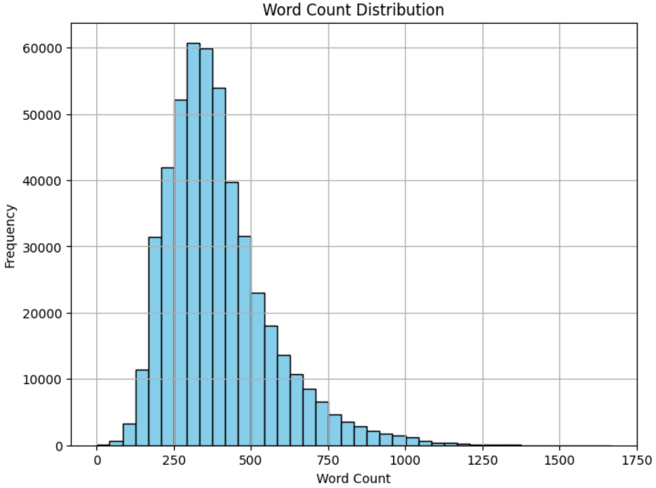
Phase 2

New Dataset: “AI vs Human Text”

- Around 500K essays
- Consists of AI and written by Human.

Visualization of Concatinated Datasets





Phase 2: Analysis

Logistic Regression

```
Accuracy: 0.8900710184756877

Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.93      0.91     92152
           1       0.88      0.82      0.85     54852

    accuracy                           0.89    147004
   macro avg       0.89      0.87      0.88    147004
weighted avg       0.89      0.89      0.89    147004
```

KNN

Random Forrest

```
Training Accuracy: 99.99%
Test Accuracy: 99.28%
              precision    recall  f1-score   support

           0       0.99      1.00      0.99     92152
           1       0.99      0.99      0.99     54852

    accuracy                           0.99    147004
   macro avg       0.99      0.99      0.99    147004
weighted avg       0.99      0.99      0.99    147004
```

SVM

Phase 2: Analysis

Distil-BERT

```
Accuracy: 0.98
Precision: 0.97
Recall: 0.98
F1 Score: 0.98
```

Phase 2: Analysis

Logistic Regression:

- Good results considering the model complexity
- Good when the dataset features are well-engineered.
- Can struggle to capture complex patterns in text

Random Forest:

- Very High Accuracy, due to its ensemble nature, combining the outputs of multiple decision trees for better generalization.
- Fast Training Time
- Good for feature Selection

DistilBertForClassification

- Accuracy slightly lower than Random Forest,
- Excels in understanding contextual information
- Long Training time

Conclusion

While a Deep NN approach can capture the context of text better, it:

- Takes a lot of training time
- Computationally expensive

On the other hand, A classical ML model like Random Forest proves to be more practical and efficient as it

- Much faster in training
- Interpretable
- Generalizes well

References

- https://towardsdatascience.com/top-machine-learning-algorithms-for-classification-2197870ff501
- https://zilliz.com/learn/tf-idf-understanding-term-frequency-inverse-document-frequency-in-nlp
- https://medium.com/@abhishekjainindore24/tf-idf-in-nlp-term-frequency-inverse-document-frequency-e05b65932f1d
- https://ayselaydin.medium.com/4-bag-of-words-model-in-nlp-434cb38cdd1b
- image link
- E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, “Detectgpt: Zero-shot machine-generated text detection using probability curvature,” PMLR, https://proceedings.mlr.press/v202/mitchell23a.html (accessed Nov. 12, 2024).
- (Mitchell et al., 2023)
- M. Guo, Z. Han, H. Chen, and J. Peng, “A Machine-Generated Text Detection Model Based on Text Multi-Feature Fusion,” Notebook for the PAN Lab at CLEF 2024, 2024.
- (Guo et al., 2024)
- X. Liu and L. Kong, “AI Text Detection Method Based on Perplexity Features with Strided Sliding Window,” Notebook for the PAN Lab at CLEF 2024, 2024.
- (Liu & Kong, 2024)