1. Introduction & Background

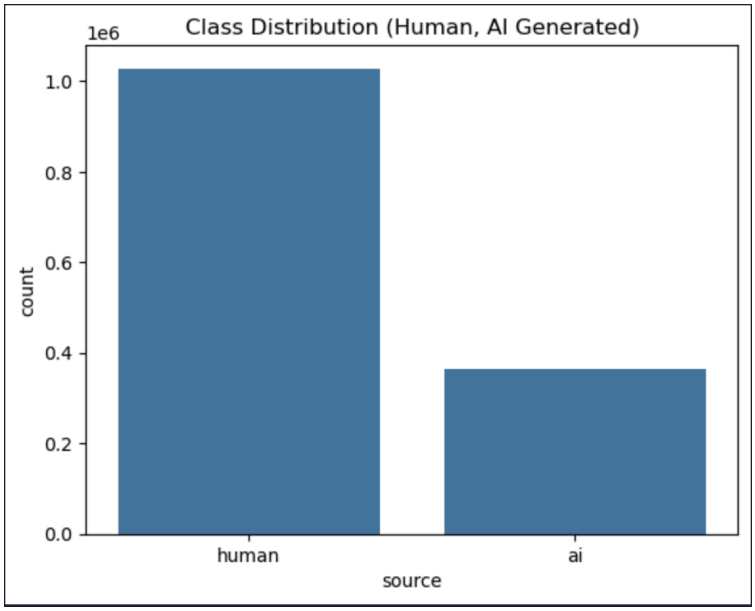
AI-generated text is becoming harder to distinguish from human writing. This is raising concerns in areas like academia, journalism, and cybersecurity. Chakraborty et al. examined the feasibility of distinguishing text produced by LLMs from those by humans through analyzing the distribution of human language versus AI's and highlighted the complexities involved in creating a reliable model [1]. Other research has shown that while current detection models can identify AI-written content, there are vulnerabilities when faced against paraphrasing tools and manual edits [2]. Weber-Wulff et al. found that many existing classifiers have high false-positive rates and are easily fooled by simple obfuscation techniques [3]. Because of these challenges, the need for more accurate and adaptable detection methods to keep up with evolving AI models is increasing. Links to the datasets we will be using:
https://huggingface.co/datasets/artem9k/ai-text-detection-pile?utm_source=chatgpt.com
This is a large scale dataset that took 990k samples of human written text from sources like Reddit WritingPromps, and AI-generated text from ChatGPT, and other GPT series
https://huggingface.co/datasets/Hello-SimpleAI/HC3?row=1
Dataset of a series of questions presented to both humans and ChatGPT and their respective responses to each of those questions
https://www.kaggle.com/datasets/sunilthite/llm-detect-ai-generated-text-dataset/data
Dataset containing both AI generated essays and human written essays

2. Problem Definition

Large language models, such as OpenAI's GPT series, can produce text that is virtually indistinguishable from that written by humans. While this is impressive, it also raises concerns about issues like academic integrity and misinformation. The challenge lies in developing reliable detection methods to AI-generated text to keep information-sharing true and safe in this digital age, and this will be the focus of our project. We will leverage advanced machine learning algorithms learned in the class to create a classification model capable of identifying AI-generated content across various contexts.

3. Methods

Data Processing With Tokenization:
We used scikit-learn's TfidfVectorizer for tokenization to transform the text data into numerical features. With this, we converted raw text data into a matrix, and determined the weights of each word by how commonly they appeared (lower weights for common ones, higher weights for rarer ones). We were able to reduce noise in the data through stop word removal and limited the features down to the top 10,000.



We got our data from a dataset found on Hugging Face, which took 990k samples of text from sources like Reddit, WritingPromps, and ChatGPT. The above graph shows the distribution between the number of human-written text compared to AI-generated text. As you can see, there's a much bigger volume of human-written text, so we took into consideration this imbalance can skew our model to predict "human" more simply because it's the majority class. We have noted that going forward we can use techniques like class weighting to address this concern.

Supervised Learning With Logistic Regression:

For our machine learning algorithm, we decided to fit a logistic regression model based on the tokenized text training samples to perform binary classification on whether each text sample was human or AI-generated. We used the logistic regression model from the scikit-learn package with a maximum of 1000 iterations.

4. Results & Discussion

Quantitative Metrics:

`Accuracy: 0.8932`

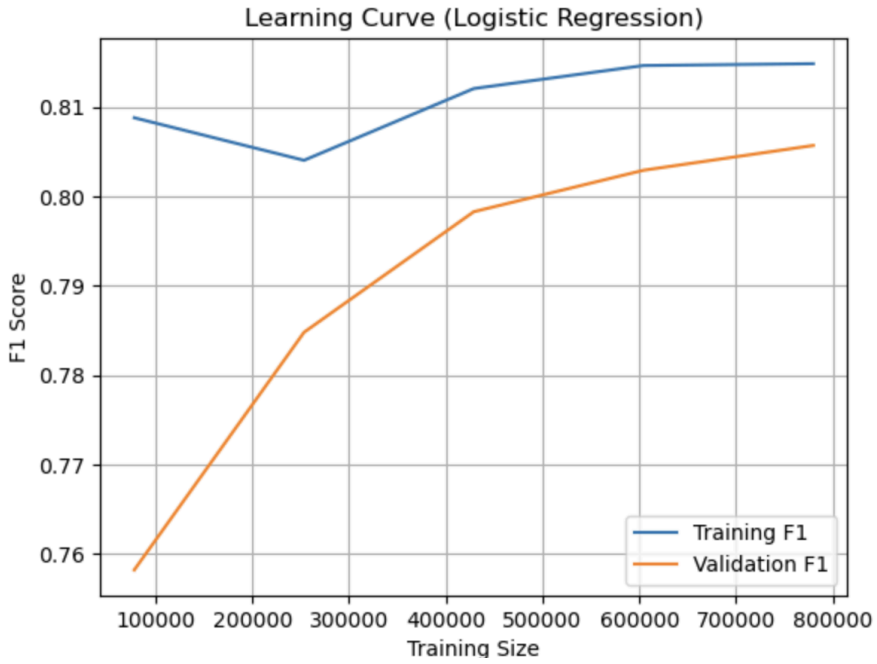
Accuracy is the ratio of correct predictions to the total number of predictions. In other words, it is the sum of the true positives and negatives divided by all categories (TP, TN, FP, FN). Accuracy is a good metric to measure model performance when the classes are balanced (they both have q similar numbers of samples), however, in this case they are not (human text is the majority class). Hypothetically, if this model classified text as "human-written" 100 % of the time, the accuracy would still be high – equal to the proportion of "human-written" data points in the dataset – even though the model fails to detect AI text.

`F1 Score: 0.8932`

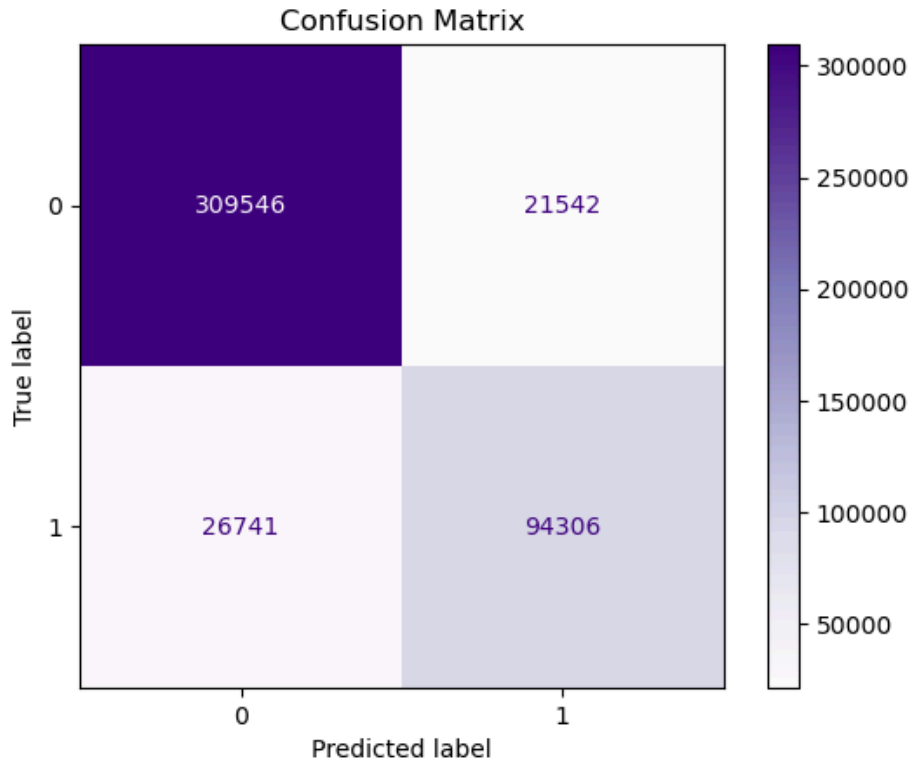
F1 Score is a good measure of model performance when the classes in the dataset are imbalanced, because it considers false positives (in precision) and false negatives (in recall). Unlike accuracy, which can be high even if the model only predicts the majority class, F-1 score mitigates this by penalizing when the model fails prediction of the minority class. A high F1 score indicates that the model maintains a good balance between precision and recall, meaning it effectively detects both human written and AI generated text without bias toward the majority class.

```
AUC: 0.9534
```

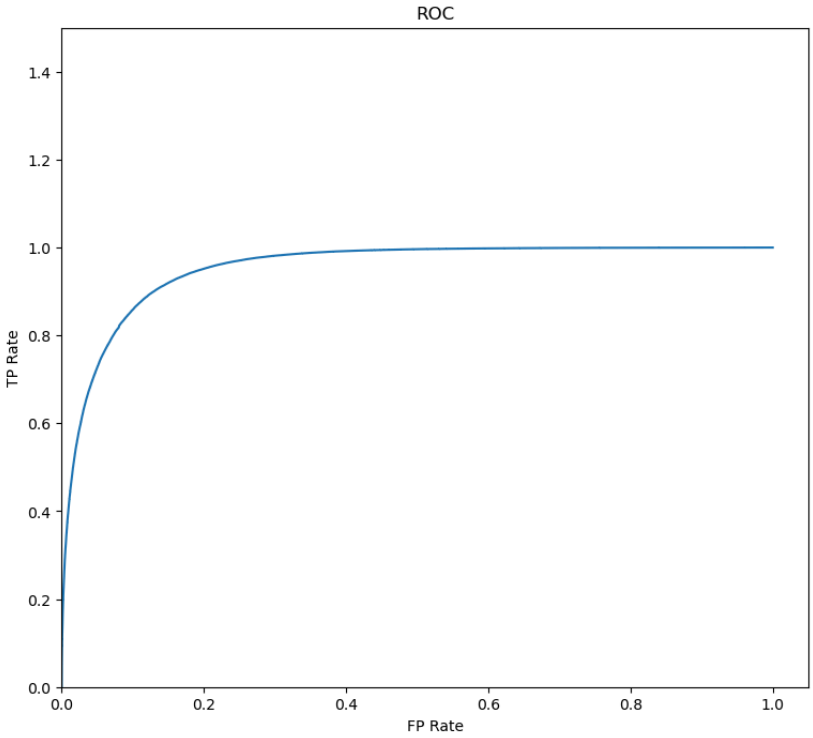
The area under the curve (AUC) is from the Receiver Operating Characteristic (ROC) curve. The x-axis is the false positive rate, and the y-axis is the true positive rate. The ROC curve visualizes the trade off between the FP and TP rates at various thresholds. An AUC of 0.5, represented by a diagonal line, is a random classifier. An AUC of 1.0 is one of the perfect classifiers. An AUC below 0.5 is a poor classifier, meaning it is worse than a random classifier. An AUC of 0.934 indicates the model has a strong ability to predict whether text is human-written or AI generated. The higher the AUC, the better the model is at minimizing false positives (incorrectly classifying human text as AI text) and maximizing true positives (correctly classifying AI text).



The blue line represents the model's performance on training data, and the orange line represents the model's performance on validation data. With smaller training sizes, there is a high score of the training data and a lower score of the validation data, indicating overfitting (common when training size is small). As the training size increases, generalization increases indicated by the decreasing gap between the training and validation lines. In other words, as the training size increases, the model becomes better at generalizing and performs better on unseen data.



The confusion matrix evaluates the performance of a classification model by showing the number of correct and incorrect predictions for each class. Predicted 1 and true 1 is a true positive, predicted 0 and true 0 is a true negative, predicted 1 and true 0 is a false positive, and predicted 0 and true 1 is a false negative. Accuracy, precision, recall, and F1-score can be calculated from the values in the confusion matrix.



This ROC curve indicates that the model has a high true positive rate and a low false positive rate.

Analysis of Model:

The logistic regression model performed well for the task of AI text detection, as can be seen by strong metrics such as a high accuracy, F1 score, and AUC. This performance can be attributed to effective practices during data preprocessing such as using TfidfVectorizer for tokenization, which weighted words based on frequency, reduced noise, and limited features to the top 10,000.

The learning curve shows that as the training size increased, the model's performance on validation data improved, demonstrating the model's improved generalization abilities. A big reason the model performed well is the size of the dataset (990k samples), which provided enough data for the model to learn important patterns.

The use of a logistic regression model is another component that resulted in high performance. Logistic regression is suitable for classification tasks that have a high dimensional feature space. The model works well on high-dimensional data because it assigns independent weights to each feature and doesn't fall into the "curse of dimensionality." Additionally, logistic regression creates a linear decision boundary in the feature space which works well for text classification. LR also provides probability scores rather than just binary classifications.

The narrowing gap between training and validation scores in the learning curve further indicates that the model successfully overcame initial overfitting and developed strong generalization capabilities, which is essential for detecting AI text across various contexts. Despite the class imbalance in the dataset (more

human-written than AI-generated samples), the model maintained robust performance metrics, suggesting effective capture of distinctive patterns between the two classes.

Next Steps:

1) Improvements in feature engineering
    a) Utilizing semantic analysis
    b) N-Gram patterns
    c) Explore word embeddings, which represents text in a dense vector space which helps with capturing more nuanced linguistic patterns
2) Testing the model on text examples that pose challenges to existing AI generated text detectors such as text modified with paraphrasing tools or manual edits. This will assess the model's robustness against the obfuscation techniques discussed in the introduction.
3) Evaluate other models, specifically the transformer-based models BERT and RoBERTa) to compare performance against logistic regression
4) Finding additional ways to address class imbalance

References

1. S. Chakraborty, A. S. Bedi, S. Zhu, B. An, D. Manocha, and F. Huang, "On the Possibilities of AI Generated Text Detection," arXiv preprint arXiv:2304.04736, 2023.
2. V. S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang, and S. Feizi, "Can AI-Generated Text be Reliably Detected?," arXiv preprint arXiv:2303.11156, 2023.
3. D. Weber-Wulff, A. Anohina-Naumeca, S. Bjelobaba, and L. Waddington, "Testing of detection tools for AI-generated text," International Journal for Educational Integrity, vol. 19, no. 1, pp. 1 25, 2023

Gantt Chart

https://onedrive.live.com/personal/11b55a31d9fd767e/_layouts/15/Doc.aspx?sourcedoc=%7B65de8d7a-6007-4eed-9a49c47e369d0d9c%7D&action=default&redeem=aHR0cHM6Ly8xZHJ2Lm1zL3gvYy8xMWI1NWEzMWQ5ZmQ3NjdlL0VYcU4zbVVIWU8xT21rbkVmamFkRFp3QnU1OTFtNkU5V0stX3FWTjZnQ1ZtV2c&slrid=a9c783a1-70c3-8000-38e921595102300a&originalPath=aHR0cHM6Ly8xZHJ2Lm1zL3gvYy8xMWI1NWEzMWQ5ZmQ3NjdlL0VYcU4zbVVIWU8xT21rbkVmamFkRFp3QnU1OTFtNkU5V0stX3FWTjZnQ1ZtV2c_cnRpbWU9Nm9RMmQ3cFMzVWc&CID=1c5f34cd-229d-4582-b3a0-47823fe8ddff&_SRM=0:G:61