Detection of AI-Generated Text for Essay Competitions

Abstract

The widespread adoption of large language models like GPT (Generative Pre-trained Transformer) has facilitated the generation of text that closely mimics human writing, raising concerns about plagiarism and academic integrity. This project aims to develop a specialized system capable of detecting AI-generated text, particularly in the context of essay submissions to academic competitions. By leveraging advanced natural language processing techniques and pre-trained machine learning models, including XLNet, GPT-2.0, and DistilBERT, we conduct a comprehensive exploration encompassing two phases of fine-tuning. The initial phrase optimizes each model's performance with the most appropriate datasets and features whereas the subsequent phase refines the models within the specific domain of academic essays. Through meticulous evaluation and comparison, we select the most suitable model for integration into our detection system. By combining cutting-edge techniques with rigorous experimentation, we seek to enhance the credibility and fairness of academic assessments while promoting a culture of originality and integrity in scholarly work.

1. Introduction

1.1 Problem Statement

In today's digital era, the proliferation of advanced AI technologies has led to heightened concerns regarding plagiarism detection and academic integrity. Essay competitions, which serve as platforms for evaluating original thought and scholarly excellence, are particularly vulnerable to submissions containing AI-generated content that resembles human writing. To address this challenge, this project aims to develop a specialized AI text detection system tailored to identify such submissions.

For the scope of this study, the solution will focus on detecting AI-generated essays submitted to academic competitions - typically around 1000 words - to tailor our detection system to the unique characteristics and requirements of such competitions, optimizing its accuracy and effectiveness.

1.2 Literature Review

Ruixiang et al. (2023) examined potential detection features for Large Language Model (LLM) generated text, encompassing statistical disparities, linguistic patterns, and fact verification. Initially, the authors tackled the detection task by employing traditional classification algorithms, such as TF-IDF combined with a logistic regression model, and other traditional algorithms like support vector machine and random forest models. This approach was advantageous for its interpretability. Subsequently, the researchers explored deep learning approaches, specifically leveraging language models like pretrained BERT models. These models were fine-tuned using a meticulously curated dataset consisting of generated-text pairs. Notably, Rodriguez et al. demonstrated that even in situations with limited available resources, robust performance could be achieved by utilizing a few hundred labeled authentic and synthetic texts specific to the domain, without the need for complete information about the LLM text generation pipeline. Therefore, our proposed project adopted a similar methodology, aligning with the flow of Rodriguez et al.'s research.

Moreover, while studies have explored LLM-generated text detection, there is a need for specialized detectors tailored to the essay competition domain, incorporating domain-specific features and writing patterns. Efforts will also be made to incorporate explainability and interpretability of models, enabling users to understand the rationale behind the model's decisions and providing meaningful feedback for improving academic writing, as suggested by Jawahar et al. (2023).

By leveraging established research frameworks and methodologies, this project plans to utilise traditional machine learning models as a baseline benchmark while delving into the potential of pretrained deep learning models, fine-tuned on both authentic and synthetic text datasets for an essay competition. In addition to this approach, the team aims to address data size limitations by initially finetuning the model using text from a broader contextual background, thereby enhancing the model's adaptability and robustness. Lastly, to address the identified research gaps stated in the reviewed literature, this project aims to develop a specialized, robust and explainable AI-generated text detector tailored to the essay competition domain.

2. Data Collection and Exploration

2.1 Data Source

Textual data from Kaggle, Hugging Face and six prestigious essay competitions were collected and

processed to support the development and evaluation of machine learning models. The data was broadly categorized into general text and real competition essay submissions, each contributing uniquely to the study.

The general text category included a dataset from Kaggle, featuring 29,145 samples of student essays and GPT(Curie)-generated essays on car-free cities. On the other hand, the Wiki Introduction dataset, sourced from Hugging Face, consisted of 150,000 pairs of Wikipedia introductions and their AI-generated versions. Both datasets were utilized for preliminary finetuning of all candidate models, among which the best combination of modeling features will be chosen for subsequent finetuning.

Under the competition essay category, an essay dataset included 50 past winning essays from five global competitions matched with GPT-3.5/4.0 generated content of similar lengths. This balanced dataset with a sum of 100 samples will be further random sampled in pairs for model finetuning and overall model evaluation. Furthermore, the test dataset comprised 25 pairs of essays from the John F. Kennedy "Profile in Courage" Essay Contest (2000-2021), alongside their GPT-4.0 generated counterparts, were utilized in deployment evaluation stage for ultimate deployment and product rollout.

2.2 Exploratory Data Analysis

A comprehensive exploratory data analysis was conducted to discern the linguistic patterns in AI-generated (AIG) text versus human-written text. Our analysis primarily used histograms to visualize distributions across several textual characteristics, as well as word clouds to highlight the prominence of words in the datasets coming from three sources: Kaggle data, Wiki introduction data, and domain-specific competition essay data (new_essay data).

As shown in Figure 3, on average, Kaggle data predominantly ranges from 150 to 600 words per essay. Wiki data often comprises 100-250 words. Notably, the new_essay dataset has been meticulously crafted to maintain an average length of 1000 words per essay that most resembles essay submissions in real competitions. Given the varying word lengths across these datasets, patterns related to essay length will be normalized to a per-word count basis to mitigate biases stemming from differences in text lengths in subsequent analysis and feature engineering.

Analyzing the sentence structure as depicted in Figure 4, the variation (standard deviation) of sentence length is compared between AIG and non-AIG texts, with AIG texts typically displaying less variation. This consistency across all datasets could be indicative of algorithmic constraints or stylistic limitations inherent in AI-generated content. Zooming into the word level, Figure 5 demonstrates that AI-generated texts generally (both in the Kaggle and new_essay datasets) feature longer words. This observation suggests a potential tendency of AI models to select syntactically complex words.

In terms of vocabulary usage, Figure 6 displays no distinct patterns regarding unique word usage between AIG and non-AIG texts. Despite the conventional understanding that humans typically exhibit more variation in word usage, the feature concerning word uniqueness is not considered in this study.

An analysis of stop words in Figure 8 revealed that AI-generated texts tend to use fewer stop words, a characteristic that could be leveraged for AI text detection. The adjective use, as indicated in Figure 9, was similarly lower in AI-generated texts, echoing with the broader trends observed in stop word usage.

Lastly, the word cloud plot in Figure 10 showed that adverbs like "even", "often", "well" are used repetitively in both AI generated and human generated text. However, AI texts exhibited a distinctive use of certain adverbs such as "moreover", "particularly", and "additionally" more so than human texts, suggesting these could be potential markers for AI involvement.

The patterns revealed through this analysis suggest practical pathways to distinguishing AI-generated text. The reduced variability in sentence length and distinct lexical choices, such as the frequent use of specific adverbs and stop words, are notable characteristics of AI-generated content. These findings could shed light on the subsequent feature engineering and model development capable of effectively identifying AI involvement in text generation.

3. Methodology

3.1 Data Pre-processing and Feature Engineering

In the data pre-processing stage, several essential steps are implemented to clean and prepare the dataset for AI-generated text detection. Firstly, duplicate entries are removed to ensure data integrity and eliminate redundancy. Text entries incorrectly labeled as both AI-generated (1) and non-AI-generated (0) are excluded to maintain consistent labeling.

In the feature engineering phase, drawing insights from the exploratory data analysis (EDA), four key features are extracted from the pre-processed text data to enrich the dataset and provide insightful analysis for AI-generated text detection:

1. *Sentence length variation* (standard deviation): This feature signifies the variation in sentence complexity and language proficiency within each text sample.
2. *Mean word length*: Providing an indication of the complexity of word usage.

3. *Percentage of stop words per text* (word count): Reflecting sentence structure, syntax, and language fluency, this feature offers valuable insights into the composition of the text.
4. *Percentage of adjectives per text* (word count): This feature serves as an indicator of rich description, expressive language, and subjective interpretation within the text samples.

By incorporating these features, we aim to capture meaningful linguistic characteristics of the text data and enhance the model's capability to detect the AI-generated text.

Once the features are extracted, all text is converted to lowercase to standardize the text format and facilitate uniform processing. Punctuation marks are removed from the text to focus solely on the textual content. Stop words, which are common words like "the", "and", "is" are removed to reduce noise and improve the relevance of the remaining words. Finally, any extra space in the text is eliminated to ensure uniformity and consistency in the dataset. These data cleaning steps are crucial for enhancing the quality and usability of the dataset, setting the foundation for effective model training and evaluation in the AI-generated text detection system.

3.2 Dataset Preparation

For preliminary finetuning, while Kaggle dataset offers a comprehensive view of sentence structure, valuable for detecting structural patterns in essays, its limitation lies in its singular focus on a specific topic, potentially constraining its generalization capacity. For Wikipedia dataset, although it showcases sophistication in writing, its content is confined to introductions rather than argumentative writing, which may limit the range of linguistic patterns available for the model to capture.

To address these limitations and ensure a more comprehensive training process, we propose a hybrid approach. By combining elements of both the Kaggle and Wikipedia datasets, we can leverage the structural insights from Kaggle while benefiting from the broader linguistic diversity offered by Wikipedia. This hybrid dataset will provide a more balanced and robust training environment for our models, enhancing their ability to generalize across various essay topics and linguistic styles. Subsequently, the efficacy of this hybrid dataset is evaluated by comparing its performance against that of the Kaggle dataset, which furnishes a complete essay structure.

For further finetuning, 30 pairs of essays (AIG and Non-AIG text) are randomly sampled from the new_essay dataset, whereas the remaining 20 pairs are used for model selection through both phases of the model selection.

Lastly, the test dataset is prepared for the deployment test. The dataset preparation process is summarized in Figure 1 below.

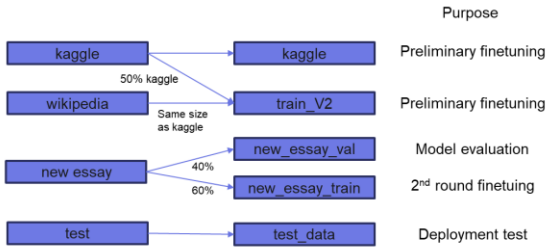


Figure 1 Summary of Dataset Preparation and Usage

3.3 Model

Three prominent NLP models, DistilBERT, XLNet, and GPT-2.0, are proposed for comparison against a baseline model, Logistic Regression.

3.3.1 Logistic Regression (Baseline Model):

Logistic Regression with Word2Vec embedding serves as a foundational baseline model for this study. Leveraging Word2Vec embeddings, which capture semantic relationships between words, the model can discern key features in the text that distinguish between AI-generated and human-written text. While not as sophisticated as transformer-based models, logistic regression provides a benchmark for comparison and can offer valuable insights into the discriminative power of more complex models.

3.3.2 DistilBERT

DistilBERT, a compact and efficient variant of BERT, is chosen as a suitable model for detecting AI-generated essays and preventing plagiarism in competitions. BERT, renowned for its advanced language understanding capabilities, employs bidirectional context and a transformer architecture to comprehend word meaning within sentences. DistilBERT is developed through a process known as knowledge distillation, where a smaller model is trained to replicate the behavior of the larger BERT model. Despite having 40% fewer parameters, DistilBERT retains approximately 97% of BERT's performance on various benchmark tasks. This reduced model size enables faster inference and efficient memory usage, making DistilBERT an optimal choice for detecting AI-generated text and identifying instances of plagiarism in essay competitions. Furthermore, DistilBERT offers scalability advantages, as it can be easily deployed and utilized on systems with limited computational resources or memory availability, without compromising significantly on performance.

3.3.3 XLNet

XLNet, specifically designed for language understanding tasks is also a suitable candidate for detecting AI generated content. Unlike traditional autoregressive models, XLNet employs a permutation language modelling objective that allows for consideration of dependencies among all permutations of input tokens and bidirectional context learning, overcoming limitations related to unidirectional pretraining. This innovative approach enables XLNet to capture long-range dependencies between words in both forward and backward directions, leading to enhanced contextual understanding and improved performance on a wide range of natural language processing tasks. Moreover, XLNet's robustness to out-of-distribution text, achieved through permutation language modeling, bidirectional context understanding, and adaptive learning, allows it to mitigate biases present in AI-generated content. The flexibility and effectiveness of XLNet make it a promising choice for developing robust and accurate AI-generated text detection systems tailored for academic integrity applications.

3.3.4 GPT-2.0

GPT-2.0 has been chosen as another candidate model that is open source and provides perspective from decoder techniques. Its pre-training on a vast corpus of 8 million web pages offers exposure to diverse linguistic patterns and styles, fostering proficiency in understanding and generating human-like text. This extensive training data equips GPT-2.0 with the ability to detect deviations from typical human language patterns that may indicate AI generation. Furthermore, GPT-2.0's autoregressive language modeling architecture facilitates the generation of coherent and contextually relevant responses, enhancing its capacity to discern subtle differences in text. Its proficiency in identifying whether an essay maintains a consistent narrative and logical progression of ideas strengthens its capability to distinguish between human-generated and AI-generated content.

3.4 Model Finetuning

The team proposes two phases of fine-tuning. The preliminary phase focuses on enhancing model adaptability by exploring different dataset combinations and evaluating the impact of including meta-features. This approach aims to overcome data size limitations and diversify the model's exposure to linguistic patterns and topic contexts, laying a robust foundation for subsequent fine-tuning stages.

3.4.1 Preliminary Finetuning

In the initial stage of model training and evaluation, we use a logistic regression model with word2vec embeddings on both the Kaggle dataset and the hybrid dataset train_v2. We evaluate its performance using the validation dataset new_essay_val, focusing on precision and recall as primary metrics. Precision is prioritized to minimize false positives while maximizing true positives, ensuring precise identification of relevant instances. Meanwhile, recall provides a comprehensive assessment of the model's ability to capture all relevant instances within the dataset. The model exhibiting superior precision and recall serves as the baseline performance benchmark,

Once the baseline performance metrics are established, each of the three pretrained models undergoes hyperparameter fine-tuning, aiming to minimize validation loss following a 7-3 train-test split. Each model is fine-tuned across four scenarios, comprising combinations of the Kaggle dataset and the hybrid dataset train_v2, with and without the inclusion of four meta-features. The optimal dataset and feature combination for each model are determined based on their precision and recall performance on the new_essay_val dataset. Below is a summary of the preliminary fine-tuning process.

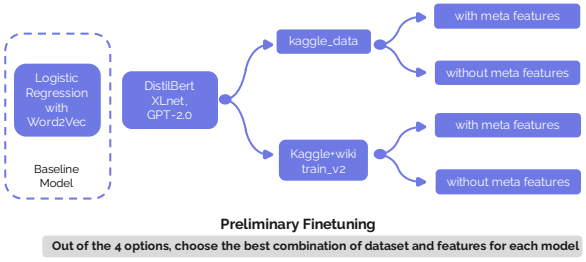


Figure 2 Summary of Preliminary Finetuning Procedures

3.4.2 Domain Adaption

To enhance the model performance on essay competition domains (with approximate 1000-word limits), we further refine the three best-performing models using the new_essay_train data. By training on domain-specific data, these models can more effectively capture the underlying linguistic patterns unique to essay writing styles, structures, and content commonly found in essay competitions.

The refined models undergo a holistic evaluation that encompasses detection performance, as well as implementation metrics such as model training effort and interpretability. By considering factors such as ease of training, interpretability of results, and computational efficiency, we gain insights into the practicality and

feasibility of deploying each model in real-world scenarios. The final model, selected based on superior performance, interpretability, and ease of implementation, undergoes a final deployment test with test_data. This test validates the model's effectiveness in real-world settings, instilling confidence in its ability to deliver reliable results in practical applications.

4. Result Discussion

4.1.1 DistilBERT

A summary of results for DistilBERT preliminary finetuning is described in Table 1.

The pretrained model has equivalent performance as the benchmark performance from Logistic regression. After further finetuning, the model in general experiences an uplift in both recall and precision. It is to be highlighted that when the model is trained on train_v2 without meta features, the recall actually dropped drastically. This may suggest potential limitations in DistilBERT's ability to effectively train on textual data alone, highlighting the importance of including meta-features. The better balance observed between precision and recall for models trained with meta-features further supports this notion. In contrast, models trained without meta-features tended to prioritize either precision or recall, indicating a lack of comprehensive understanding or context when relying solely on textual data. The incorporation of meta features in the same dataset helped strike a better balance between precision and recall by providing additional context and discriminative information, enabling the model to make accurate positive predictions while also capturing a higher proportion of true positive instances.

The impact of topic diversity on model training varies depending on the inclusion of meta features. When meta features are included, the hybrid dataset train_v2, which combines data from Kaggle and Wikipedia sources, generally led to better overall performance compared to the Kaggle dataset alone. The diverse topics and comprehensive nature of the hybrid dataset likely provided a more representative sample of the underlying data distribution, allowing the model to generalize better and capture patterns more effectively. However, with the absence of meta features, the introduction of diverse topics may have the opposite impact. Model 5 improves the precision from model 3 at a greater cost of recall. This absence of meta features could limit the model's ability to capture additional context or information that is crucial for maintaining high recall.

As a result, model 4 with the best balance of precision and recall is chosen for domain adaptation at a later stage.

Table 1 Results for DistilBERT Preliminary Finetuning

SN	Configuration	Training Data	Precision	Recall
0	LR w/ Word2Vec & meta features	train_v2 (28738 rows × 5 cols)	0.593	0.8
1	Pretrained	-	0.56	0.7
2	w/ meta features	Kaggle (27340 rows × 5 cols)	0.895	0.85
3	w/o meta features	Kaggle (27340 rows × 1 col)	0.72	0.9
4	w/ meta features	train_v2 (28738 rows × 5 cols)	0.9474	0.9
5	w/o meta features	train_v2 (28738 rows × 1 col)	1	0.4

The comparison of model 4's performance before and after domain adaptation is summarized in Table 2. Following additional fine-tuning on the 60 rows of 1000-word essays, precision further increased to 1 indicating perfect precision in detecting AI-generated essay. Moreover, the model maintained a high recall of 0.9. This improvement highlights the effectiveness of domain adaptation in enhancing the model's performance within the specific domain of academic essays.

Table 2 Performance of DistilBERT in Domain Adaptation

SN	Configuration	Training Data	Precision	Recall
4	w/ meta features	train_v2 (28738 rows × 5 cols)	0.9474	0.9
6	Model 4 finetuned	new_essay_train (60 rows × 1 col)	1	0.9

4.1.2 XLNet

A summary of results of XLNet preliminary finetuning is described in Table 3

It is noted that despite all the different combinations, XLnet consistently yields a high recall, which indicates its robust ability to effectively capture and classify AI generated text. However, this often comes at the expense of precision especially when the model is not trained on appropriate data. The experimental findings underscore the crucial role of dataset diversity in enhancing XLNet's performance. While the Kaggle dataset alone offers limited benefits for model fine-tuning, integrating data from both Kaggle and Wikipedia sources in train_v2 significantly improves precision. This finding suggests while XLNet tends to excel with out-of-distribution text, a dataset that purely focuses on one topic may not fully leverage its capabilities. Exposing to diverse topics and contexts enables the model to generalize better and make more robust predictions across various domains and scenarios.

To further analyse the effectiveness of incorporating meta-features in the hybrid train_v2 dataset, the addition of

meta-features resulted in further improvement in precision compared to the model without meta-features, while maintaining a high recall. meta-features offer a way to augment XLNet's capabilities by providing additional context, domain knowledge, and complementary information, thereby improving its performance, generalization, and adaptability, especially in scenarios with limited data or specialized domains.

As a result, model 4 with the highest precision while maintaining high recall is chosen for further finetuning.

Table 3 Results for XLNet Preliminary Finetuning

SN	Configuration	Training Data	Precision	Recall
0	LR w/ Word2Vec & meta features	train_v2 (28738 rows × 5 cols)	0.593	0.8
1	Pretrained	-	0.545	0.9
2	w/ meta features	Kaggle (27340 rows × 5 cols)	0.571	1
3	w/o meta features	Kaggle (27340 rows × 1 col)	0.556	1
4	w/ meta features	train_v2 (28738 rows × 5 cols)	0.947	0.9
5	w/o meta features	train_v2 (28738 rows × 1 col)	0.818	0.9

The comparison of model 4's performance before and after domain adaptation is summarized in Table 4. Following additional fine-tuning on the 60 rows of 1000-word essays, both precision and recall further increase to 1 and 0.95 respectively, demonstrating the effectiveness of domain adaptation.

Table 4 Performance of XLNet in Domain Adaptation

SN	Configuration	Training Data	Precision	Recall
4	w/o meta features	train_v2 (28738 rows × 1 col)	0.947	0.9
6	Model 4 finetuned	new_essay_train (60 rows × 1 col)	1	0.95

4.1.3 GPT-2

A summary of results for GPT-2.0 preliminary finetuning is described in Table 5. Without further training, the pretrained model has no predictive power, with 0 in both precision and recall. However, after an initial round of finetuning with general context data, both precision and recall exhibit noticeable improvements. All combinations in general have a better performance than the baseline model (Logistic regression with word2vec).

Comparing datasets, the hybrid dataset (a mix of Wikipedia introductions and Kaggle essays) generally achieves a better balance of precision and recall when trained on the same features as GPT-2.0. While the Kaggle dataset may yield higher precision or recall individually, it often fails to achieve equivalent results to the hybrid dataset when predicting on different topics. Models trained on the hybrid dataset tend to demonstrate superior overall performance in both aspects. This suggests that the hybrid dataset provides a more comprehensive and diverse training environment, enabling the model to capture patterns and differences more effectively. Despite the constraint of limited linguistic patterns in Wikipedia introductions, the model benefits from the diverse topics inherent in the hybrid dataset. This observation implies that GPT-2.0's performance is less confined by linguistic patterns and more influenced by the breadth of topics covered.

Within the same dataset, the impact of meta features varies. For the Kaggle dataset, introducing meta features boosts precision but reduces recall. This is likely because meta features provide additional context for the model's predictions, enhancing precision by aiding accurate positive predictions. However, this selectivity can lead to a decrease in recall as the model may miss some true positive instances. Conversely, for the hybrid dataset, the incorporation of meta features unexpectedly decreases precision without enhancing recall. It's possible that the diverse topics in the hybrid dataset already provide sufficient contextual information for the GPT-2.0 model, rendering additional meta features redundant or even detrimental to performance. Moreover, irrelevant or noisy meta features could potentially introduce noise and degrade model performance.

Table 5 Results for GPT-2.0 Preliminary Finetuning

SN	Configuration	Training Data	Precision	Recall
0	LR w/ Word2Vec & meta features	train_v2 (28738 rows × 5 cols)	0.593	0.8
1	Pretrained	-	0	0
2	w/ meta features	Kaggle (27340 rows × 5 cols)	0.938	0.750
3	w/o meta features	Kaggle (27340 rows × 1 col)	0.760	0.950
4	w/ meta features	train_v2 (28738 rows × 5 cols)	0.818	0.900
5	w/o meta features	train_v2 (28738 rows × 1 col)	0.900	0.900

The comparison of model 5's performance before and after domain adaptation is summarized in Table 6 Following additional fine-tuning on the 60 rows of 1000-word essays, both precision and recall further increase to 0.95, demonstrating the effectiveness of domain adaptation.

Table 6 Performance of GPT-2.0 in Domain Adaptation

SN	Configuration	Training Data	Precision	Recall
5	w/o meta features	train_v2 (28738 rows × 1 col)	0.900	0.900
6	Model 5 finetuned	new_essay_train (60 rows × 1 col)	0.95	0.95

However, it's crucial to acknowledge that the above findings regarding the performance of the three models require validation on a more extensive dataset. The current analysis is constrained by the limited sample size of the test data from new_essay_val, comprising only 40 rows. Therefore, to ensure the robustness and reliability of the conclusions drawn, further examination on a larger and more diverse dataset is imperative.

4.2 Model Evaluation

A summary of the model evaluation is described in the Table below. When considering the various criteria for evaluating DistilBERT, XLNet, and GPT-2.0, each model demonstrates strengths and weaknesses across different aspects. GPT-2.0 stands out as the final choice due to its balance of predictive power and deployment speed. Despite its large size, GPT-2.0 requires relatively faster fine-tuning efforts compared to XLNet, making it more feasible for adapting to specific tasks efficiently. While deployment speed may be slower than DistilBERT, GPT-2.0's moderate deployment speed is still manageable, especially considering the simpler preprocessing of articles which requires no meta features, unlike the other two models. Although interpretability and scalability are complex across all models, GPT-2.0's excellent predictive performance and reasonable trade-offs in other areas position it as the optimal choice for the task at hand.

Table 7 Summary of Model Comparison

Criteria	DistilBERT	XLNet	GPT-2.0
Prediction ability	Precision:1 Recall: 0.9	Precision:1 Recall: 0.95	Precision:0.95 Recall: 0.95
Model Size	Smaller	Large	Largest
Finetuning Effort	Fastest (~10min / epoch)	Slowest (~70 min/ epoch)	Relative faster (~55 min/ epoch)
Deployment Speed	Faster	Slowest Large size and required additional layer for meta features	Moderate no additional layer to obtain meta features
Interpretability	Complex	Complex	Complex
Scalability and Resource Usage	Scalable with less resource requirement	Less scalable due to large size	Less scalable due to large size

5. Model Deployment

The final stage of the AI text detector project involves deploying the best-performing model, GPT-2.0, into a user-friendly web application. This deployment strategy is designed to make the tool accessible to a wide audience, including essay competition organizers and participants and educators.

Prior to deployment, a pre-deployment test was conducted on the refined GPT-2.0 model using test data, resulting in precision and recall scores of 1 and 0.56 respectively. The discrepancy in recall performance could be attributed to the more advanced capabilities of GPT-4.0 in learning human writing styles. GPT-4.0, trained on a substantially larger dataset, likely possesses enhanced abilities to replicate human writing patterns. Moreover, as the model was primarily trained on text generated by GPT(Curie) or GPT-3.5, it may have less generalization ability to the advancements made in GPT-4.0. This will be further discussed in the limitations and future work.

The front-end of the web application will provide a clean, simple interface that allows users to easily upload or paste text for analysis. The application will then process the input text and pass to model for prediction and provide immediate feedback in the form of predicted label as well as the probability that indicates the overall likelihood of AI involvement in their submitted text.

6. Limitation and Future Work

We encountered several limitations that impacted the scope and effectiveness of our analyses and outcomes.

One significant limitation was the computational constraints we faced. The process of running our models, particularly during the fine-tuning phase, required extensive GPU resources and time. As a result, we were compelled to limit the number of tuning epochs. It inevitably hindered our ability to explore a wider array of parameters and may have prevented us from achieving the optimal performance in our models. This reflects a common challenge in data-intensive commercial products where the trade-off between model complexity and practical feasibility must be carefully managed. To improve on this, future iterations of the project could leverage more efficient computational strategies such as distributed computing or the use of more advanced hardware. In addition, implementing more sophisticated model optimization techniques such as model pruning might allow us to achieve higher efficiency without sacrificing performance.

Another major limitation was related to the quality and quantity of the data. Our project utilized domain-specific data comprising past winning essays in competitions.

While this data was valuable, the lack of a larger, more diverse data set, including lower quality essay submissions, restricted our ability to understand the full spectrum and breadth of human writing. It in turn impacts the generalizability of our model across different styles and formats of text. Furthermore, the current models predominantly utilize training data generated by GPT-3.5. To enhance detection capabilities and accommodate evolving generative technologies, it is imperative to incorporate a more substantial number of samples from GPT-4.0, which represent the latest advancements in AI text generation. The issue of data quantity and quality is a prevalent concern in machine learning projects, where the data constraint can significantly influence the robustness and accuracy of the outcomes. To mitigate this limitation, future research could focus on aggregating a larger dataset from more diverse sources. Engaging in partnerships with educational institutions and prestigious essay contests could provide access to a broader array of text samples, enhancing the model's learning and its ability to generalize across various text types.

In addition, while GPT-2.0 has shown considerable efficacy in identifying AI-generated texts, it may not match the capabilities of more advanced models like GPT-4.0, particularly as some of the AI-generated texts in our datasets were produced using GPT-4.0. The advancements in GPT-4.0 could potentially introduce linguistic patterns that GPT-2.0 is less equipped to detect due to its older architecture and training data. Despite the superior performance that might be expected from using GPT-4.0 as a detection model, our access to this more advanced model was limited due to its proprietary and costly nature. To address this limitation, future research could explore the feasibility of accessing more advanced models through funded research opportunities. Employing transfer learning techniques could potentially narrow the performance gap between GPT-2.0 and GPT-4.0, enabling the older model to better adapt to the complexities introduced by the newer generation of language models. This approach would allow the project to extend its capabilities without incurring prohibitive costs, ensuring both economic and technological scalability.

Finally, our project faced limitations in its ability to display detailed insights into plagiarism for our users. Our tool provides a probabilistic estimate of whether a submission may contain plagiarized content rather than pinpointing specific sections of the text. This limitation reduces the utility of our tool for users who require detailed, actionable insights, such as contestants in writing competitions who need to understand precisely which parts of their text may need revision. Improving this feature could involve the integration of natural language understanding techniques that focus on semantic analysis, which could enhance the precision of plagiarism detection. Moreover, incorporating feedback loops where users can validate or refute the flags raised by the system could help in fine-tuning the algorithm, increasing both its accuracy and reliability.

Together, these limitations delineate the challenges we faced during the project, impacting both the process and the final outcomes. Addressing these challenges in future work will be crucial for advancing the project's capabilities and for refining the tools we have developed to distinguish between AI-generated and human-written text. The proposed improvements aim to enhance the effectiveness of our methodologies and extend the applicability of our findings to a wider range of real-world applications.

7. Conclusion

Throughout this research, we have explored three pre-trained models and tailored them to effectively recognize the characteristics and patterns of AI-generated essays through finetuning on multiple datasets. We can improve the precision and recall, hence the reliability of the plagiarism detection system in essay competition settings, thereby supporting fair and credible assessments. This research lays a foundational framework for future advancements in the detection of AI-generated text, paving the way for more secure and trustworthy competition environments.

References

Wu, J., Yang, S., Zhan, R., Yuan, Y., Wong, D.F. and Chao, L.S., 2023. A survey on LLM-generated text detection: Necessity, methods, and future directions. *arXiv preprint arXiv:2310.14724*.

Tang, R., Chuang, Y.N. and Hu, X., 2024. The Science of detecting llm-generated text. *Communications of the ACM*, *67*(4), pp.50-59.

Orenstrakh, M.S., Karnalim, O., Suarez, C.A. and Liut, M., 2023. Detecting llm-generated text in computing education: A comparative study for chatgpt cases. *arXiv preprint arXiv:2307.07411*.

Walters, William H.. "The Effectiveness of Software Designed to Detect AI-Generated Writing: A Comparison of 16 AI Text Detectors" Open Information Science, vol. 7, no. 1, 2023, pp. 20220158. https://doi.org/10.1515/opis-2022-0158

Rodriguez Juan, Todd Hay, David Gros, Zain Shamsi, and Ravi Srinivasan. 2022. Cross-Domain Detection of GPT-2-Generated Technical Text. *In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* Association for

Computational Linguistics, Seattle, United States, 1213–1233.

Appendix 1. Figures for EDA

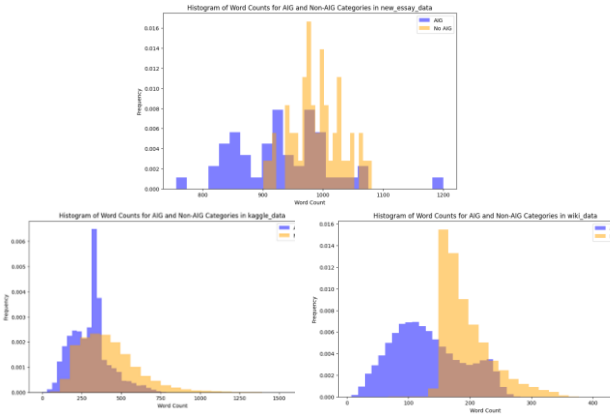


Figure 3 Histogram of Word Count in AI-generated (AIG) and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

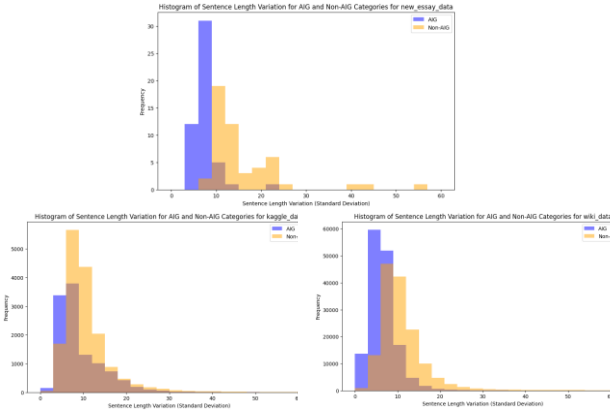


Figure4 Histogram of Sentence Variation in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

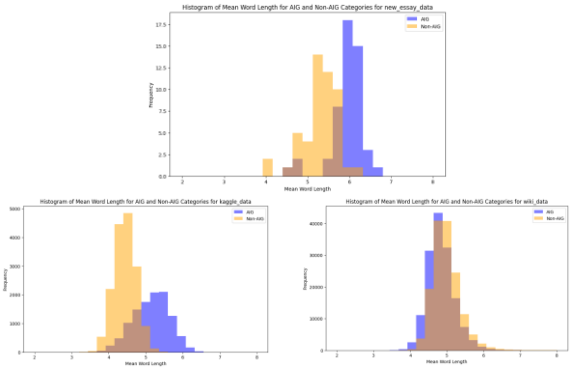


Figure 5 Histogram of Word Length for AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

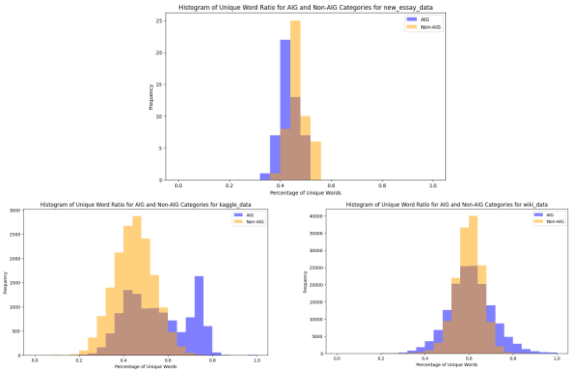


Figure 6 Histogram of Unique Word Frequency in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

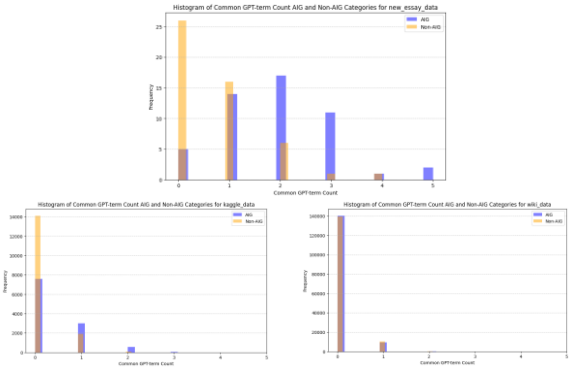


Figure 7 Histogram of Common GPT Terms Freq. in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

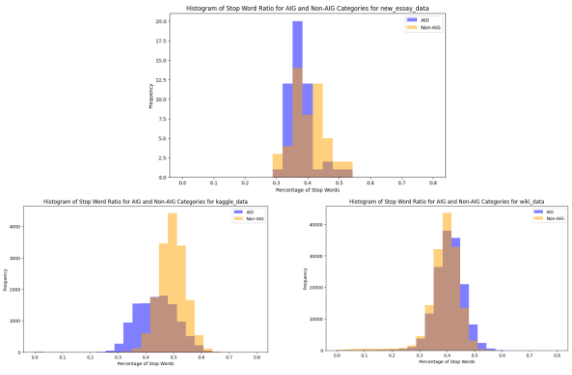
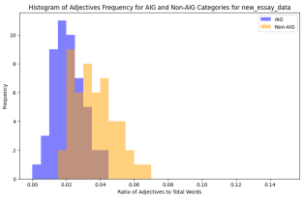


Figure 8 Histogram of Stop Word Frequency in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively



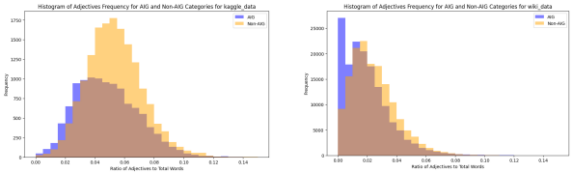


Figure 9 Histogram of Adjectives Frequency in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

Figure 10 Word Cloud for Adjectives in AIG and Non-AIG Texts in New_Essay, Kaggle, Wiki Datasets respectively

Appendix 2. Code for Github

https://github.com/Gallifrey-SG/BT5153_team_project