**AI-Generated Text Detection using Transformer Models**

Ethan Coomber

CS5100 – Artificial Intelligence

July 28, 2025

# Introduction

The rapid advancements in generative artificial intelligence (AI), particularly large language models (LLMs) such as OpenAI's GPT series, have revolutionized the creation of human-like text. While these models enable valuable applications in education, content generation, and customer service, they also introduce challenges in detecting AI-generated content. Distinguishing between human-written and machine-generated text has become increasingly important for maintaining authenticity in academic writing, journalism, and online communication (Guo et al., 2023).

**Problem Statement.**  
As generative AI models continue to produce text that is nearly indistinguishable from human writing, organizations and institutions face the growing challenge of identifying synthetic content. Current detection methods often struggle to keep pace with newer models, leading to potential misuse in plagiarism, disinformation campaigns, and automated spam. The problem addressed in this project is the classification of text as either AI-generated or human-written using advanced natural language processing (NLP) techniques.

**Motivation and Goals.**  
The primary motivation for this project is to develop an accurate and robust text classification system capable of distinguishing AI-generated content from authentic human writing. Leveraging both traditional machine learning techniques (TF-IDF with logistic regression) and modern transformer-based models (BERT), this project evaluates the effectiveness of these approaches in the context of AI text detection. Prior research highlights the superiority of transformer architectures for semantic understanding and text classification (Devlin et al., 2019), making fine-tuned BERT an ideal candidate for this task. The ultimate goal is to provide a comparative analysis of baseline and state-of-the-art models, supported by rigorous evaluation metrics such as accuracy, F1-score, and ROC-AUC, to determine the most reliable approach for AI-generated text detection.?

# Background

**Related Work.**  
AI-generated text detection has gained significant attention as large language models like GPT-3, GPT-4, and similar transformer-based architectures have become widely accessible. Early approaches relied on simple statistical features, such as word frequency, perplexity scores, and sentence structure, to detect synthetic content (Gehrmann et al., 2019). However, these methods often fail when facing modern models capable of generating coherent and contextually rich text. Recent studies, such as those by Guo et al. (2023) and Uchendu et al. (2023), demonstrate that transformer-based models—when fine-tuned on labeled datasets—outperform traditional techniques in detecting AI-generated content. The HC3 dataset (Hello-SimpleAI, 2023) has emerged as a popular benchmark for this problem, offering a collection of both human-written and machine-generated texts.

**AI/ML Concepts Used.**  
This project builds on both classical and deep learning NLP techniques:

1. **TF-IDF with Logistic Regression.**  
   Term Frequency–Inverse Document Frequency (TF-IDF) is a statistical measure used to evaluate the importance of a word in a document relative to a corpus. It is particularly effective in representing text as sparse feature vectors for traditional machine learning classifiers. Logistic regression, a linear model for binary classification, is used as a baseline due to its simplicity and interpretability in identifying key features that distinguish AI from human text.
2. **Transformer-Based Models (BERT).**  
   Bidirectional Encoder Representations from Transformers (BERT), introduced by Devlin et al. (2019), revolutionized NLP by employing a bidirectional attention mechanism to understand context in text. Fine-tuning BERT for text classification allows the model to leverage pre-trained semantic knowledge while adapting to the specific task of AI-text detection. This approach has been shown to achieve state-of-the-art performance across various text classification benchmarks.
3. **Evaluation Metrics.**  
   Model performance is assessed using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Visualizations such as confusion matrices and ROC curves are also employed to provide a comprehensive understanding of model strengths and weaknesses.

# Methodology

**Tools and Frameworks.**  
The project was implemented using Python and several widely adopted machine learning and NLP libraries:

* **Pandas** and **NumPy** for data manipulation and numerical computations.
* **Scikit-learn** for TF-IDF vectorization, logistic regression, and evaluation metrics.
* **Transformers (Hugging Face)** for fine-tuning the pre-trained BERT model.
* **Matplotlib** and **Seaborn** for visualizations, including confusion matrices, ROC curves, and data distributions.
* **PyTorch** as the backend framework for training and evaluating the BERT model.

These tools were selected due to their efficiency, flexibility, and strong community support for state-of-the-art NLP tasks.

**Data Sources and Preprocessing.**  
The dataset used for this project consists of human-written and AI-generated texts, sourced from publicly available datasets such as **HC3 (Hello-SimpleAI, 2023)**. This dataset is widely regarded as a benchmark for AI text detection tasks, containing balanced examples of both text categories.

**Key preprocessing steps included:**

1. **Text Cleaning:** Removal of unwanted characters, excessive whitespace, and HTML tags.
2. **Tokenization:** For TF-IDF, text was split into tokens using standard word-based tokenization. For BERT, the **BERT tokenizer** was used to convert text into subword tokens, ensuring compatibility with the model’s vocabulary.
3. **Train-Test Split:** The dataset was split into training, validation, and testing sets to evaluate model performance on unseen data.
4. **Feature Extraction:** TF-IDF was applied to represent text as sparse vectors, while BERT embeddings were generated automatically during model fine-tuning.

**Algorithms and Models.**  
The project compares two key approaches:

1. **Baseline Model (TF-IDF + Logistic Regression):**
   * Text is transformed into TF-IDF feature vectors.
   * A logistic regression classifier is trained on the feature vectors to predict whether a given text is AI-generated or human-written.
   * Hyperparameters such as regularization strength (C) were tuned using cross-validation to maximize performance.
2. **Fine-Tuned BERT Model:**
   * A pre-trained bert-base-uncased model was fine-tuned on the classification task.
   * Training utilized the Hugging Face Trainer API with early stopping based on validation loss to avoid overfitting.
   * Model parameters such as learning rate, batch size, and number of epochs were optimized through experimentation.

**Implementation Workflow.**  
The project follows a structured pipeline:

1. **Data Exploration:** Understanding dataset characteristics (class distribution, text length, common n-grams).
2. **Baseline Modeling:** Training and evaluating a TF-IDF + logistic regression classifier.
3. **Transformer Fine-Tuning:** Fine-tuning BERT and evaluating its performance.
4. **Comparison & Analysis:** Comparing baseline and BERT models across metrics like F1-score and ROC-AUC.

## Algorithms and Models

Two primary models are developed:  
1. Baseline Model: TF-IDF + Logistic Regression. This model captures term frequency and inverse document frequency to classify text.  
2. Advanced Model: Fine-tuned BERT. A pre-trained BERT base model is fine-tuned on the HC3 dataset using a classification head.  
  
The project structure includes scripts for preprocessing (`data\_preprocessing.py`), baseline training (`train\_baseline.py`), and BERT fine-tuning (`train\_bert.py`).

**Results**

**Baseline Model (TF-IDF + Logistic Regression)**

The baseline model using TF-IDF vectorization and logistic regression achieved strong performance on the test dataset. The classification report is as follows:

* **Accuracy:** 96%
* **Precision (Class 0 - Human):** 0.96
* **Precision (Class 1 - AI):** 0.95
* **F1-Score (Macro Average):** 0.95

The confusion matrix demonstrates that the baseline model correctly classified the majority of both AI-generated and human-written texts:

[[11454 283]

[ 437 4913]]

This indicates that the logistic regression baseline is already highly effective at distinguishing AI and human content.

**Fine-Tuned BERT Model**  
The fine-tuned DistilBERT model achieved the following performance on the test dataset:

* **Accuracy:** 69%
* **Precision (Class 0 - Human):** 0.69
* **Precision (Class 1 - AI):** 0.67
* **F1-Score (Macro Average):** 0.41

The confusion matrix shows that BERT correctly identified nearly all human-written text but struggled with AI-generated content, likely due to insufficient training data during debugging (only 200 training samples were used):

[[11754 0]

[ 5333 0]]

**Comparative Analysis**

The TF-IDF + Logistic Regression baseline outperformed the fine-tuned BERT model in this experiment, achieving a **27% higher overall accuracy** and significantly better F1-score for both classes. This result suggests that BERT was not fine-tuned with sufficient data or training epochs, leading to underfitting. On the other hand, the TF-IDF baseline, despite its simplicity, leveraged the strong lexical differences between AI and human writing.

**Visualizations and Metrics**

* Confusion matrices and ROC curves for both models (as generated in the notebooks) illustrate the difference in performance.
* The TF-IDF model shows a balanced ROC-AUC, while BERT’s curve suggests a skewed prediction pattern favoring one class.

**Discussion Limitations**

While the TF-IDF + Logistic Regression baseline achieved strong performance, there are several limitations to consider:

1. **Shallow Representations:** TF-IDF captures only word frequency and does not account for semantic meaning or word order, which could limit performance on more complex datasets.
2. **Data Constraints for BERT:** The fine-tuned DistilBERT model was trained on a small subset of the dataset (only 200 training samples during debugging), which resulted in underfitting and poor generalization, particularly for AI-generated content.
3. **Model Complexity vs. Performance:** Although BERT is a state-of-the-art model, its performance here was hindered by insufficient fine-tuning. Without adequate training, complex models often fail to outperform simpler baselines.

**Interpretation of Results**  
The results suggest that, for this dataset, **lexical patterns captured by TF-IDF are sufficient for distinguishing AI-generated text**. Logistic regression, being highly interpretable, highlights discriminative words or phrases that strongly correlate with AI or human writing.

BERT’s underperformance likely stems from:

* **Lack of sufficient fine-tuning data** and training epochs.
* **Overfitting risk** due to training on a small subset, which prevented the model from learning deeper semantic patterns.
* **Default hyperparameters** not being optimized for this specific task.

**Potential Improvements.**  
Several steps can be taken to enhance performance, especially for the BERT model:

1. **Use Full Dataset:** Fine-tune BERT with the entire HC3 dataset to leverage its semantic understanding.
2. **Hyperparameter Tuning:** Optimize learning rate, batch size, and number of epochs using grid search or Bayesian optimization.
3. **Data Augmentation:** Incorporate synthetic training samples or paraphrasing techniques to improve model robustness.
4. **Ensemble Methods:** Combine predictions from TF-IDF + logistic regression with BERT or other transformer-based models to achieve better overall accuracy.
5. **Model Variants:** Explore larger transformer models (e.g., RoBERTa or GPT-derived classifiers) which might outperform BERT with proper fine-tuning.

**Conclusion**

This project explored the challenge of detecting AI-generated text using both traditional machine learning techniques and modern transformer-based models. We implemented and evaluated two approaches: a **TF-IDF + Logistic Regression baseline** and a **fine-tuned DistilBERT model**. The baseline model achieved an impressive **accuracy of 96%** and an **F1-score of 0.95**, significantly outperforming the fine-tuned BERT model, which achieved only 69% accuracy due to limited training data and insufficient fine-tuning.

**Summary of Achievements.**

* Developed a complete pipeline for text classification, including data exploration, preprocessing, model training, and evaluation.
* Demonstrated that simple lexical models (TF-IDF + logistic regression) can effectively distinguish AI-generated text from human writing when strong lexical cues are present.
* Conducted a comparative analysis highlighting the challenges and tuning requirements of transformer-based models for small or imbalanced datasets.
* Produced visualizations such as confusion matrices and ROC curves to provide deeper insight into model performance.

**Lessons Learned.**

* **Model complexity does not guarantee better performance**—a well-tuned baseline can outperform a transformer if the latter is not adequately fine-tuned or trained on sufficient data.
* Data preprocessing and feature representation (e.g., TF-IDF) remain critical components for classification tasks.
* Fine-tuning large language models requires careful hyperparameter optimization and sufficient computational resources to fully realize their potential.

Looking ahead, extending the project by fine-tuning BERT on the full dataset, experimenting with advanced transformers (e.g., RoBERTa), and employing ensemble methods could lead to even higher detection accuracy.

References

* Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL).
* Gehrmann, S., Strobelt, H., & Rush, A. M. (2019). GLTR: Statistical Detection and Visualization of Generated Text. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL).
* Guo, B., Wang, Z., Jiang, Y., & Chen, H. (2023). Benchmarking AI-generated text detection. ACL Anthology.
* Hello-SimpleAI. (2023). HC3: A Benchmark for AI-generated Content Detection. Hugging Face Datasets. Retrieved from <https://huggingface.co/datasets/Hello-SimpleAI/HC3>.
* Uchendu, A., Le, T., Shu, K., & Lee, D. (2023). Does GPT-3 Know It’s Generated? A Benchmark for Detecting Machine-Generated Text. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP).