ARTIFICIAL INTELLIGENCE
PROJECT REPORT
De-AI Cipher: Decoding the Language of Machines

Submitted by: Hardik Sharma (TUid: 916292428)

1 Abstract

The sudden advancement in Natural Language Processing (NLP) models, such as GPT and other large language models (LLMs), has revolutionized text generation by producing coherent and contextually relevant content that mimics human writing. While these capabilities enhance various domains like content creation, education, and customer service, they also pose challenges concerning authenticity, plagiarism, misinformation, and accountability. **This project addresses the critical task of distinguishing between AI-generated and human-written text through a comprehensive evaluation of linguistic and statistical metrics**. Using datasets tailored to this purpose, multiple machine learning models, including traditional algorithms like Logistic Regression and advanced models like Multilayer Perceptron and XGBoost, were tested for their performance. The results highlight the effectiveness of models like XGBoost and Multilayer Perceptron in achieving high accuracy and F1 scores across various datasets. This work provides a robust framework for enhancing transparency and ethical AI usage, contributing to a secure and accountable digital ecosystem.

2 Introduction

AI models, such as OpenAI's GPT and similar large language models (LLMs), can now produce coherent, contextually relevant, and persuasive text that often mimics human writing. While these capabilities have transformed various domains, they have also raised significant concerns.

Distinguishing between AI-generated text and human-written text has become crucial for maintaining trust, ethical standards, and accountability in communication. Whether it is identifying AI-generated academic essays, detecting machine-generated fake news, or ensuring transparency in automated responses, the ability to discern the source of text is essential in today's digital age. However, current tools have notable limitations that call for the development of more robust, accurate, and scalable methods.

To address this, this project leverages a comprehensive suite of linguistic and statistical metrics to build a robust and scalable detection method. These metrics include Perplexity, Entropy (character-wise and word-wise), Burstiness, Stylometric Analysis, N-gram (bi-gram and tri-gram) Analysis, Semantic Coherence, and Repetition Detection (repeating words, ratio, and n-gram counts). Additional measures encompass Syntactic analysis, Psycholinguistic features, Readability scores (e.g., Flesch-Kincaid), Sentiment polarity, Interrogative content analysis, Cognitive load indicators, and various character-level features, such as special character and punctuation counts, as well as error patterns like spelling and grammatical errors. To enhance reliability, multiple machine learning algorithms are employed to identify the optimal model for distinguishing AI-generated from human-written text. This project is both timely and critical, contributing to a more transparent and accountable use of AI in modern communication.

3 Related Works

The task of distinguishing between AI-generated and human-generated text has gained significant attention in recent years, leading to the development of numerous methodologies and metrics tailored to this purpose. Early AI-generated texts were relatively easy to identify due to their lack of coherence and limited vocabulary. However, modern LLMs, such as OpenAI's GPT series, have achieved a level of fluency and contextual understanding that often makes their outputs indistinguishable from human writing. Researchers have explored various approaches to address this issue, including watermarking, statistical and stylistic analysis and machine learning analysis [2] but each method presents unique challenges, especially as AI models continue to improve in mimicking human writing styles. A study introduced SeqXGPT, a method based on convolution and self-attention networks utilizing log probability lists from white-box LLMs as features for sentence-level AI-generated text detection. Experimental results show that previous methods struggle in solving sentence-level AIGT detection, while our method not only significantly surpasses baseline methods in both sentence and document-level detection challenges but also exhibits strong generalization capabilities [23]. Another paper leveraged state-of-the-art machine learning models such as RoBERTaBase, RoBERTa-Large, and SVM, to detect subtle differences in language patterns, stylistic features, and semantic nuances. This paper presents evidence that helps to support the challenge that human-generated sentences can be differentiated from sentences generated from GPT-3.5-Turbo [24].

For this project, I reviewed and analyzed related works corresponding to each metric that I decided to work with, incorporating details about them into the methodology section to provide a comprehensive understanding of the field. In addition to discussing these works in depth, I have compiled supplementary materials, including links to relevant research papers, and tools, to aid in deeper exploration. These resources have been curated and are available in the GitHub repository LLMMetricsResearch (https://github.com/hrdikshrma/LLMMetricsResearch). This repository serves as a consolidated hub for understanding the landscape of distinguishing AI-generated text from human-written text and highlights the progression of this field.

4 Methodology

4.1 Proposed Approach

The diagram (Figure 1) illustrates a workflow for this project.

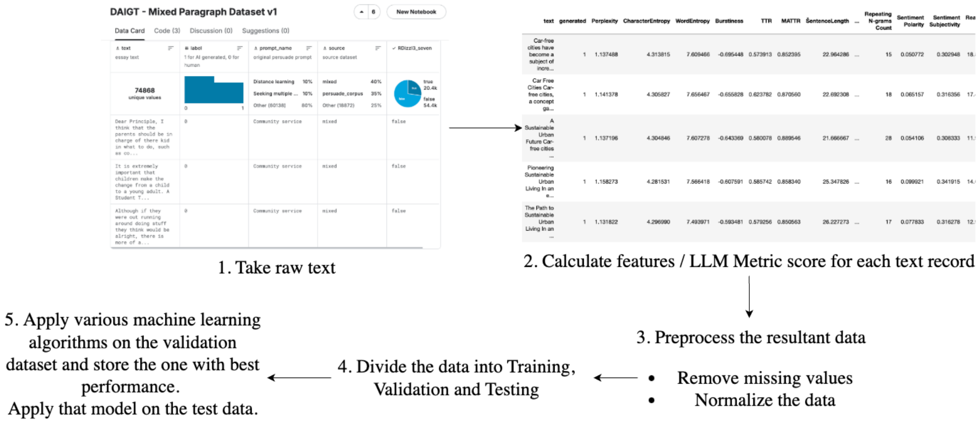
1. The process begins with loading raw text data. The dataset contains textual data such as paragraphs or sentences and a label for whether it is AI-generated or human-written. It can also some metadata such as its source and other characteristics.

2. For each record in the text dataset, specific features or metrics are calculated.

3. Once features are extracted, the data is preprocessed to ensure it is suitable for machine learning. Preprocessing involves: (1) removing missing values,i.e., eliminating or imputing records with incomplete feature sets. (2) normalizing the data, i.e., scaling features to a uniform range or distribution to improve model performance.

4. The dataset is split into three subsets namely, training set, for model training, validation set, for tuning hyperparameters and testing set, for evaluating the final model's performance on unseen data.

5. Various machine learning models are trained and evaluated using the validation set to determine the best-performing algorithm. Examples of algorithms that could be applied include random forests, support vector machines (SVMs), or deep learning models. The best-performing model is then used to make predictions or classifications on the test data.



DAIGT - Mixed Paragraph Dataset v1

text	label	prompt_name	source	RDizzl3_seven
essay text	1 for AI generated, 0 for human	original persuade prompt	source dataset	
74868 unique values	0 1	Distance learning 10% Seeking multiple ... 10% Other (60138) 80%	mixed 40% persuade_corpus 35% Other (18872) 25%	true 20.4k false 54.4k
Dear Principle, I think that the parents should be in charge of there kid in what to do, such as co...	0	Community service	mixed	false
It is extremely important that children make the change from a child to a young adult. A Student T...	0	Community service	mixed	false
Although if they were out running around doing stuff they think would be alright, there is more of a...	0	Community service	mixed	false

1. Take raw text

text	generated	Perplexity	CharacterEntropy	WordEntropy	Burstiness	TTR	MATTR	ŠentenceLength	...	Repeating N-grams Count	Sentiment Polarity	Sentiment Subjectivity	Rea
Car-free cities have become a subject of incre...	1	1.137488	4.313815	7.609466	-0.695448	0.573913	0.852395	22.964286	...	15	0.050772	0.302948	18.
Car Free Cities Car-free cities, a concept ga...	1	1.141378	4.305827	7.656467	-0.655828	0.623782	0.870560	22.692308	...	18	0.065157	0.316356	17.
A Sustainable Urban Future Car-free cities ...	1	1.137196	4.304846	7.607278	-0.643369	0.580078	0.889546	21.666667	...	28	0.054106	0.306333	11.
Pioneering Sustainable Urban Living In an e...	1	1.158273	4.281531	7.566418	-0.607591	0.585742	0.858340	25.347826	...	16	0.099921	0.341915	14.
The Path to Sustainable Urban Living In an ...	1	1.131822	4.296990	7.493971	-0.593481	0.579256	0.850563	26.227273	...	17	0.077833	0.316278	12.

2. Calculate features / LLM Metric score for each text record

3. Preprocess the resultant data

- Remove missing values
- Normalize the data

4. Divide the data into Training, Validation and Testing

5. Apply various machine learning algorithms on the validation dataset and store the one with best performance.
Apply that model on the test data.

Figure 1: Proposed Approach

4.2 Datasets

To ensure the generalizability of the results and to avoid any potential biases in the modeling process, I tested the workflow on three diverse datasets. This approach not only aims to improve the reliability of the evaluation but also ensures the model's adaptability to various real-world scenarios.

1. **DAIGT - Catch The AI**: This data consists of different LLMs , such as: Mistral-7B(v1&v2) , Llama 70b , Falcon180b ,GPT(3.5 & 4), Claude.

   Training Records: 25969, Validation Records: 2730 and Testing Records: 2730

   Link: https://www.kaggle.com/datasets/zeyadusf/daigt-all-data-for-competition

2. **DAIGT - Mixed Paragraph Dataset v1**: All Records: 74868 unique records

   Link: https://www.kaggle.com/datasets/serjhenrique/daigt-mixed-paragraph-dataset-v1

3. **LLM - Detect AI-Generated Text Dataset**: The dataset comprises of a mixture of 28,000 student-written essays and essays generated by a variety of LLMs.

   All Records: 27340 unique records

   Link: https://www.kaggle.com/datasets/sunilthite/llm-detect-ai-generated-text-dataset

4.3 LLM Metrics

4.3.1 Perplexity

For a given text, Perplexity is a measure of how well a language model predicts the text. Lower perplexity indicates the text is more predictable or aligned with the language model's training, while higher perplexity suggests the text is harder to predict. We can calculate the perplexity of a given text using a pre-trained language model, in my case, I used *bert-base-uncased.*

AI-generated text tends to have lower perplexity because AI models are designed to minimize uncertainty and produce highly probable sequences of text. Human-written text typically has higher perplexity due to greater variability and unpredictability [1].

It is important to note that this difference in perplexity doesn't necessarily indicate quality. Lower perplexity in AI-generated text doesn't mean it is better; it simply means it is more predictable according to the model's training. Human-written text, with its higher perplexity, often contains more creativity, nuance, and unexpected insights, which are valuable qualities in many contexts. As AI models continue to advance, the gap in perplexity between AI and human-generated text may narrow, making detection and differentiation more challenging

4.3.2 Entropy

For a given text, Entropy measures the randomness or diversity of word/character usage by evaluating the probability distribution of words/characters in the text. Higher entropy indicates greater variation in word/character choice, while lower entropy suggests repetitive or predictable language. I have calculated the word-wise entropy and character-wise entropy of each text.

In most cases, human-generated text is likely to have greater entropy than AI-generated text [2]. This is because (1) Humans introduce more unpredictability and variability in their writing. (2) AI models, especially advanced ones, are designed to minimize uncertainty and produce highly probable sequences of text. (3) Human writing often includes unique insights, unexpected connections, and creative elements that increase entropy.

However, it is important to note that this can vary based on the specific AI model used, the type of text generated, and the measurement method.

4.3.3 Burstiness

For a given text, Burstiness measures the unevenness or irregularity of words across the text. If a word appears in quick bursts (closely clustered positions) rather than evenly spread, it has higher burstiness. The metric is calculated based on the intervals between occurrences of the same word.

Perplexity treats each word prediction as equally important, disregarding the bursty nature of language, where certain words or phrases occur more frequently in specific contexts. While Perplexity measures how well an AI model forecasts the next word, Burstiness goes beyond by capturing the intricate dance of words, revealing their hidden patterns and clustering [3].

A higher burstiness score indicates more word repetition relative to the vocabulary size.

4.3.4 Type-Token Ratio

For a given text, Type token ratios (TTR) are a measurement of linguistic diversity. They are defined as the ratio of unique tokens divided by the total number of tokens. This measurement is bounded between 0 and 1. If there is no repetition in the text this measurement is 1, and if there is infinite repetition, it will tend to 0. This measurement is not recommended if analyzing texts of different lengths, as when the number of tokens increases, the TTR tends to flatten. More advanced measures like Moving Average Type-Token Ratio (MATTR) may provide more reliable comparisons across different text lengths.

If the text or the document has the lowest TTR value then it has more function words than the content words [4]. Function words are the filler words of a language, such as pronouns, prepositions, and modifying verbs, that fit around the content of a sentence.

4.3.5 Moving Average Type-Token Ratio

For a given text, MATTR is calculated by choosing a window length (say 500 words) and then computing the TTR for words 1–500, then for words 2–501, then 3–502, and so on to the end of the text. The mean of all these TTRs is a measure of the lexical diversity of the entire text and is not affected by text length nor by any statistical assumptions. Further, the individual TTRs can be compared to detect changes within the text. This helps smooth out fluctuations caused by varying text lengths.

Human-written texts tend to show greater lexical diversity and vocabulary richness. AI-generated texts often exhibit lower TTR scores, indicating less variety in both tokens and structures used [5].

4.3.6 Average Sentence Length

For a given text, the average sentence length of an input text is useful for analyzing the complexity of writing style — longer sentences might indicate more complex or formal writing. It is calculated by dividing the total number of words in the text by the total number of sentences.

Human-written texts tend to have more varied sentence lengths, mixing short and long sentences for rhythm and emphasis. AI-generated text often shows more uniform sentence lengths, lacking the natural variation found in human writing.

Average sentence length should be used alongside other metrics like lexical diversity, perplexity, and burstiness for more accurate differentiation. This metric alone is not definitive. Advanced AI models can be programmed to vary sentence length. The effectiveness of this measure may depend on the specific AI model and how it is trained. Human writing styles differ greatly, so there's no one-size-fits-all threshold for sentence length. Authors in this publication stated that sentence complexity (depth) is the only category without a significant difference between humans and ChatGPT-3, as well as ChatGPT-3 and ChatGPT-4 [6].

4.3.7 Stopword Frequency

For a given text, stopword frequency calculatez the frequency of function words (e.g., prepositions, conjunctions, articles, and pronouns) in each text. Function words, often called stopwords, are essential for grammatical structure but carry less semantic meaning. The function computes the ratio of function words to the total number of words, helping analyze writing style and formality.

AI models and humans tend to use function words differently. AI-generated text often shows more consistent and predictable patterns in function word usage. Human writers typically have more varied and context-dependent use of function words. Because Large Language Models work by predicting the next word in a sentence, they are more likely to use common words like "the," "it," or "is" instead of wonky, rare words [7].

4.3.8 N-Grams Calculation

Bi-grams are consecutive pairs of words, and we identify the top 5 most frequently occurring bi-grams in the text. Tri-grams are consecutive sequences of three words. This analysis helps uncover common word pairs, which can provide insights into writing patterns or repetitive phrases. N-gram analysis can capture subtle differences in how AI and humans use context and phrase structure. Human writing typically involves more varied n-grams and creative language choices.

AI-generated texts often show more consistent and predictable n-gram patterns. Human-written texts typically exhibit more varied n-gram distributions, especially for higher values of n. AI-generated texts have been found to have a higher frequency of the same n-grams, particularly in higher n-gram ranges [1]. This increased repetition in AI text suggests that language models identify certain sequences as "safe bets" for generation.

4.3.9 Semantic Coherence

For a given text, semantic coherence measures how closely related consecutive sentences are. Semantic coherence indicates the flow and logical connection between sentences. This method can capture nuanced differences in how AI and humans maintain logical flow and connections between ideas. It is calculated by using embeddings generated from a pre-trained transformer model (in this case, I used *Sentence Transformer*).

Advanced AI models are continuously improving in generating coherent text, potentially narrowing the gap with human writing. The effectiveness of this measure alone may not be sufficient, as AI-generated text can sometimes maintain high coherence levels. The choice of the pre-trained model for generating embeddings can influence the results. A publication stated that features in coherence and consistency are significant to predict human-written and AI-generated texts. AI-generated texts have higher coherence but have lower internal consistency. Both the text generated by humans and that generated by AI can deliver semantic information, which results in machine learning models based on semantic features having limited explanatory power [8]. In another work, Fröhling and Zubiaga categorized linguistic features according to which potential weakness of AIGT they measure: (1) lack of syntactic and lexical diversity; (2) repetitiveness; (3) lack of coherence; and (4) lack of purpose [9][13]. In another research, the comparison between human and AI-generated texts reveals that while AI systems like ChatGPT can produce academically sound writing, human writers still hold the advantage of making more nuanced and sophisticated content. Human-authored texts consistently demonstrate a wider range of cohesive techniques, including more subtle forms of cohesion, such as substitution and ellipsis, that contribute to a more dynamic and flexible writing style. Human writers are also better at progressively developing their ideas, integrating new information to build on previous concepts, and enhancing the text's overall coherence. This is a key area where AI-generated texts often fall short, as they rely heavily on explicit cohesion through conjunctions and lexical repetition, making the structure feel overly linear and somewhat predictable [10].

4.3.10 POS Tagging

For a given text, Part-of-Speech (POS) tagging on a given text categorizes the counts of different POS tags into predefined categories (e.g., nouns, verbs, modifiers). POS tagging identifies the grammatical role of each word in the text (like noun, verb, adjective), and categorizing these tags helps in understanding the structure and style of the text.

The syntactic analysis using POS tags alone may not be highly effective in distinguishing AI from human text. One study found no significant differences in UPOS (Universal Part-of-Speech) tag distribution between AI-generated and human-written texts [11]. Another study presented a model that classifies human or computer-generated texts, using vocabulary richness metrics and POS label ratios to train a simple artificial neural network for Spanish classification, and some other features to build a Naïve Bayes Model for English classification. The objective is to classify texts in both English and Spanish. The results show a Macro F1 of 0.67 for the texts in English and 0.6441 for the texts in Spanish. These numbers show that the classifier can distinguish computer-generated texts from human texts using the POS features with some reliability, although it is clear that there is a lot of room for improvement [12].

4.3.11 Repetition Analysis

For a given text, word repetition analysis is performed to find word repetitions in each text. It identifies words that occur more than once and calculates the repetition ratio, which is the proportion of repeated word occurrences to the total number of words. This helps in understanding the redundancy or emphasis in the text. The word repetition ratio provides a quantitative measure that can be compared across different texts, potentially revealing differences between AI and human writing styles.

AI-generated texts often show more consistent repetition patterns, especially in phrases (n-grams) learned from the training data [13]. Human-written texts typically have more varied repetition, with both high and low repetition rates, reflecting natural thought progression and writing style. Online articles also state that one of the hallmarks of text generated by a machine learning model like ChatGPT is a certain amount of repetition. The model may repeat phrases or sentences in its output, which is unlikely to occur in text written by a human [14][15][16]. Humans can be creative with their use of language and imagery, whereas AI-generated text can be repetitive or lack originality. For example, a human might use idiomatic expressions, or write in a unique and personal style, which an AI model like ChatGPT might not be able to replicate.

4.3.12 Readability Score

For a given text, the Flesch-Kincaid readability score measures how easy a text is to read, based on the average number of words per sentence and syllables per word. A higher score indicates easier readability, while a lower score suggests the text is more complex.

Research has shown that AI-generated texts have lower Flesch-Kincaid scores than human-written texts. In one study, AI-generated articles had a mean Flesch-Kincaid score of 46, while human-written articles had a mean score of 59. This suggests that on average, AI-generated text may be more complex or less readable than human-written text [17]. Another article stated, that though AI can generate grammatically correct sentences, it often lacks a natural, engaging flow. Enhancing readability might involve simplifying complex sentences, using more conversational language, or incorporating rhetorical devices such as metaphor and analogy [20].

4.3.13 Sentiment Polarity and Subjectivity

For a given text, sentiment analysis is performed using the following 2 techniques:

- Polarity: A value between -1 and 1 that indicates the sentiment of the text. Negative values represent negative sentiment, positive values represent positive sentiment, and 0 represents neutral sentiment.
- Subjectivity: A value between 0 and 1 that indicates how subjective or opinionated the text is. Higher values represent more subjective or personal opinions, while lower values represent more factual content.

The above metrics can reveal patterns in emotional tone and objectivity that might differ between AI and human writing. AI-generated text might show more consistent sentiment patterns, while human writing could have more varied emotional expressions. Subjectivity scores might reveal differences in how personal opinions are expressed in AI vs. human writing. an article states that firstly, ChatGPT generates text based on patterns and patterns from the data it's trained on, it does not have personal experiences, so it's unlikely to include personal anecdotes or reflections. Secondly, AI model like ChatGPT may not be able to replicate the complexity of human emotions, so it might not be able to capture the nuances of sentiment or tone in text [14].

4.3.14 Interrogative Content

For a given text, interrogative content is analyzed by identifying and counting the number of questions. It uses two criteria to detect questions:

- Sentences ending with a question mark (?).
- Sentences that start with common question words or subject-auxiliary inversion patterns (e.g., "What," "Why," "Is," "Can").

Humans tend to use questions more naturally within the flow of their writing, often for rhetorical effect or to engage readers. AI may use questions more formulaically. The frequency of questions in a text could be an indicator, as human writers may use questions differently than AI models. But, this method alone is not sufficient for reliable differentiation. Advanced AI models can mimic human question patterns effectively.

4.3.15 Cognitive Verbs

For a given text, Congitive analysis is done by counting the number of cognitive words in a text. Cognitive verbs are action words associated with mental processes like thinking, analysing, evaluating, or creating. These verbs are often indicators of higher-order cognitive activity and are useful for assessing the cognitive load or complexity of the text.

4.3.16 Special Characters

For a given text, the number of special characters is also calculated. Special characters include symbols like @, #, $, %, ^, &, *, (,), _, +, =, and -. These characters are often used in technical documents, code snippets, or casual text (like social media posts).

4.3.17 Spelling Error

For a given text, the count of the number of spelling errors in each text is also analyzed. It compares each word in the text against a dictionary of correctly spelled words. Words not found in the dictionary are considered misspelled. This helps evaluate the grammatical quality of the text.

4.3.18 Grammar Errors

For a given text, the count of the number of grammatical errors in each text is analyzed using the LanguageTool library. It scans the text for grammar issues, such as incorrect verb tense, subject-verb agreement errors, or improper punctuation, and returns the total number of detected errors.

An article states that while both human and AI-generated text may contain errors, the nature of those errors is different. Humans might have typos, misspellings, and grammatical errors, while AI-generated text may have more systematic errors like repeating words or using the wrong word [14][21]. One article said that human-written text is generally more polished and free of errors, while machine-generated text may contain grammatical errors or misspellings. This is because humans have a deep understanding of language and are able to self-correct and revise their writing, while machines simply output the text that is most likely based on the data they have been trained on [22].

4.4 Machine Learning Algorithms

1. Logistic Regression: A linear model used for binary or multi-class classification that predicts probabilities using a logistic function.

2. K-Nearest Neighbor (KNN): A simple instance-based learning algorithm that classifies data points based on the majority class of their nearest neighbors.

3. SVM (Linear): A support vector machine that finds a hyperplane in a linear space to separate classes with maximum margin.

4. SVM (Polynomial): A variation of SVM that uses a polynomial kernel to model non-linear relationships between features.

5. SVM (Gaussian): An SVM using the Gaussian (RBF) kernel to handle complex non-linear relationships.

6. Naïve Bayes Classifier: A probabilistic classifier based on Bayes' theorem, assuming independence between features.

7. Decision Tree: A tree-structured model that splits data into subsets based on feature values to classify data.

8. Random Forest: An ensemble learning method that combines multiple decision trees to improve classification accuracy and reduce overfitting.
9. XGBoost: A high-performance gradient boosting framework that builds strong classifiers by combining weak learners iteratively.
10. Multilayer Perceptron (MLP): A feedforward neural network with multiple layers of neurons capable of capturing complex patterns in data.

5 Results

5.1 Dataset : DAIGT — Catch The AI

The results in Table 1 show that the Multilayer Perceptron (MLP) achieves the highest performance during validation, with an accuracy and F1-score of 0.9813, followed closely by XGBoost at 0.9806. Support Vector Machines (SVMs) with Polynomial and Gaussian kernels also perform exceptionally well, while the Naïve Bayes Classifier lags behind with the lowest accuracy and F1-score. During testing, the results of BERT (that has been used as a baseline) and the MLP model are quite comparable, demonstrating the effectiveness of our proposed approach. MLP and XGBoost exhibit superior performance, highlighting their effectiveness for this dataset, while BERT showcases strong consistency in testing.

In the models Decision Tree, Random Forrest and XGBoost (Figure 2,3 and 4), *Grammer Error* feature emerges as the most significant feature, contributing the most to the classification accuracy. *Spelling errors, readability scores*, and features like *stopword frequency and pronouns* also play secondary roles in Random Forest and Decision Tree models.

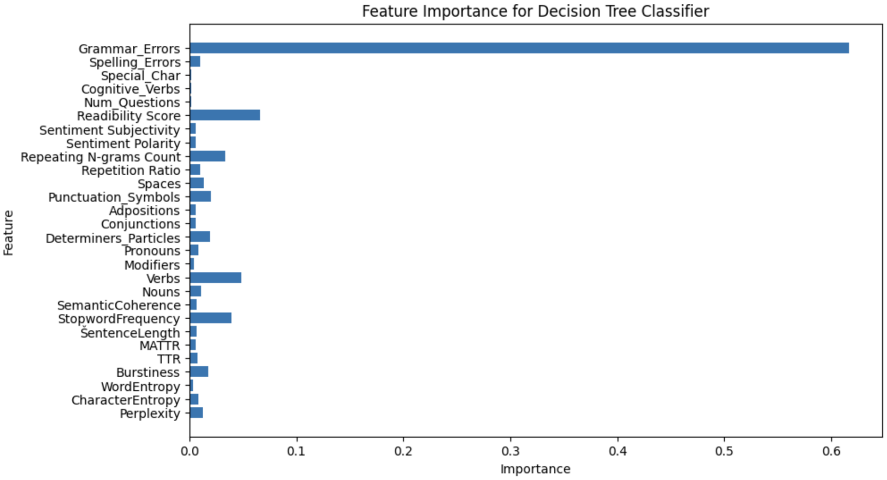


Figure 2: Feature Importance - Decision Tree Classifier

Classification Algorithms	Accuracy	F1-score
Validation		
Logistic Regression	0.9344	0.9353
K-Nearest Neighbor	0.9322	0.9335
SVM (Linear)	0.9516	0.9523
SVM (Polynomial)	0.9766	0.9766
SVM (Gaussian)	0.9722	0.9722
Naïve Bayes Classifier	0.8300	0.8426
Decision Tree	0.9498	0.9502
Random Forest	0.9326	0.9334
XGBoost	0.9806	0.9806
Multilayer Perceptron	0.9813	0.9813
Testing		
Multilayer Perceptron	0.9758	0.9758
BERT (for baseline)	0.9765	0.9770

Table 1: Performance metrics for classification algorithms on dataset (DAIGT — Catch The AI)

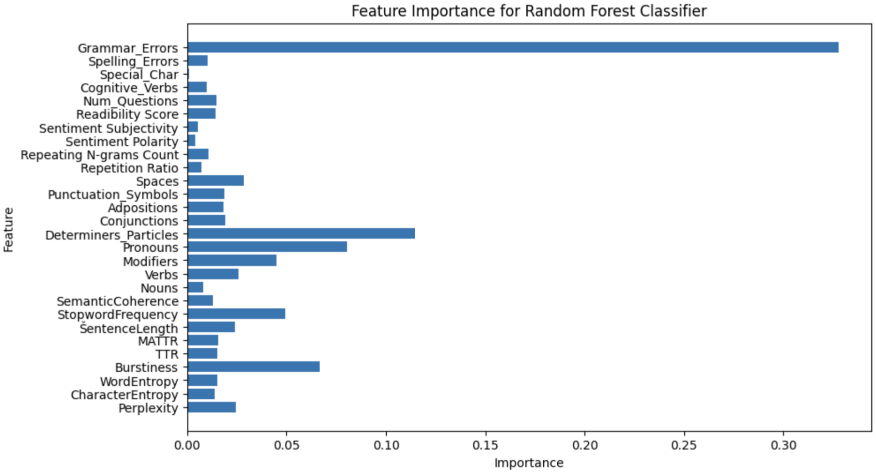


Figure 3: Feature Importance - Random Forest Classifier

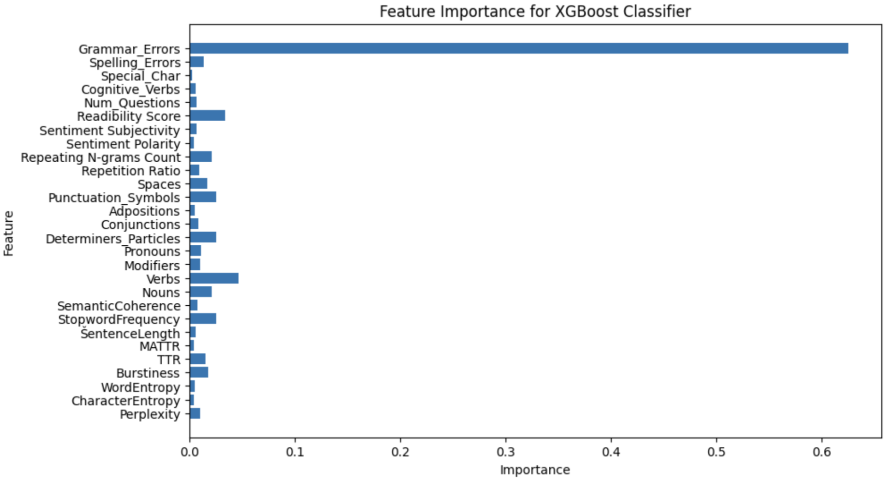


Figure 4: Feature Importance - XGBoost Classifier

5.2 Dataset: DAIGT - Mixed Paragraph Dataset v1

The results in Table 2 highlight XGBoost as the best-performing classification model, achieving a validation accuracy and F1-score of 0.8738, outperforming other models such as Logistic Regression, SVM (various kernels), Decision Trees, and Random Forest. On the testing set, XGBoost maintains strong performance with an accuracy of 0.8669 and an F1-score of 0.8665.

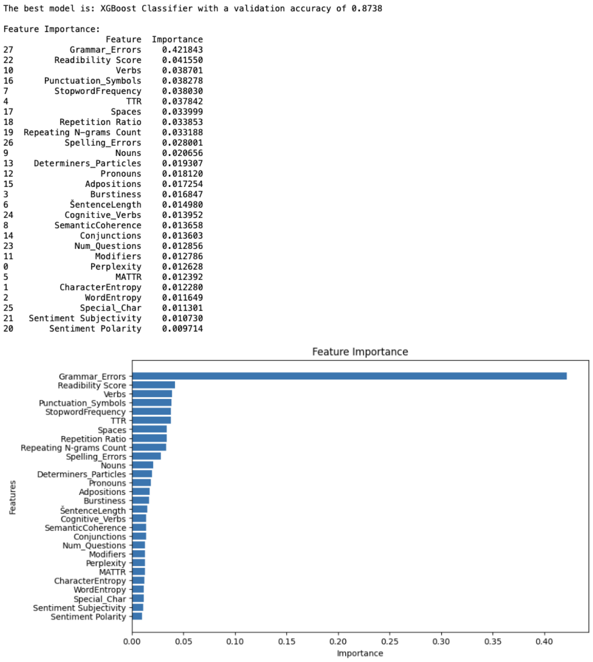
A detailed analysis of feature importance (Figure 5) reveals that *Grammatical Errors* contribute the most to the classification, with an importance score of 0.4218, making it a critical predictor for the model once again. *Readability score* (0.0416) and *verbs* (0.0387) are the next most influential features, followed closely by *punctuation symbols* (0.0383) and *stopword frequency* (0.0379). Additional features such as *repetition ratio, and repeating n-grams count* exhibit moderate importance.

Less impactful but still noteworthy features include *spelling errors, nouns, and determiners/particles*, while semantic-level features like *semantic coherence, sentence length, and burstiness* provide marginal contributions. Features related to entropy measures, such as *word entropy and character entropy*, as well as sentiment-related attributes like *sentiment polarity and subjectivity*, are among the least influential, indicating a lesser role in the classification task.

Overall, the results demonstrate that XGBoost effectively leverages a diverse set of linguistic and stylistic features, with grammar-related metrics playing a dominant role in its success. The strong performance on both validation and testing sets highlights its suitability for this classification problem.

Classification Algorithms	**Accuracy**	**F1-score**
	Validation	
Logistic Regression	0.8449	0.8446
K-Nearest Neighbor	0.7860	0.7863
SVM (Linear)	0.8540	0.8538
SVM (Polynomial)	0.8676	0.8674
SVM (Gaussian)	0.8617	0.8615
Naïve Bayes Classifier	0.7003	0.6965
Decision Tree	0.8023	0.8023
Random Forest	0.8088	0.8058
XGBoost	0.8738	0.8736
Multilayer Perceptron	0.8710	0.8698
	Testing	
XGBoost	0.8669	0.8665

Table 2: Performance metrics for classification algorithms on dataset (DAIGT - Mixed Paragraph Dataset v1)



```
The best model is: XGBoost Classifier with a validation accuracy of 0.8738

Feature Importance:
                   Feature  Importance
27          Grammar_Errors    0.421843
22         Readibility Score    0.041550
10                   Verbs    0.038701
16      Punctuation_Symbols    0.038278
7         StopwordFrequency    0.038030
4                      TTR    0.037842
17                  Spaces    0.033999
18        Repetition Ratio    0.033853
19  Repeating N-grams Count    0.033188
26         Spelling_Errors    0.028001
9                    Nouns    0.020656
13    Determiners_Particles    0.019307
12                Pronouns    0.018120
15             Adpositions    0.017254
3               Burstiness    0.016847
6           ŠentenceLength    0.014980
24         Cognitive_Verbs    0.013952
8        SemanticCoherence    0.013658
14            Conjunctions    0.013603
23           Num_Questions    0.012856
11               Modifiers    0.012786
0               Perplexity    0.012628
5                    MATTR    0.012392
1         CharacterEntropy    0.012280
2              WordEntropy    0.011649
25            Special_Char    0.011301
21   Sentiment Subjectivity    0.010730
20      Sentiment Polarity    0.009714
```

Figure 5: Feature Importance - XGBoost Classifier

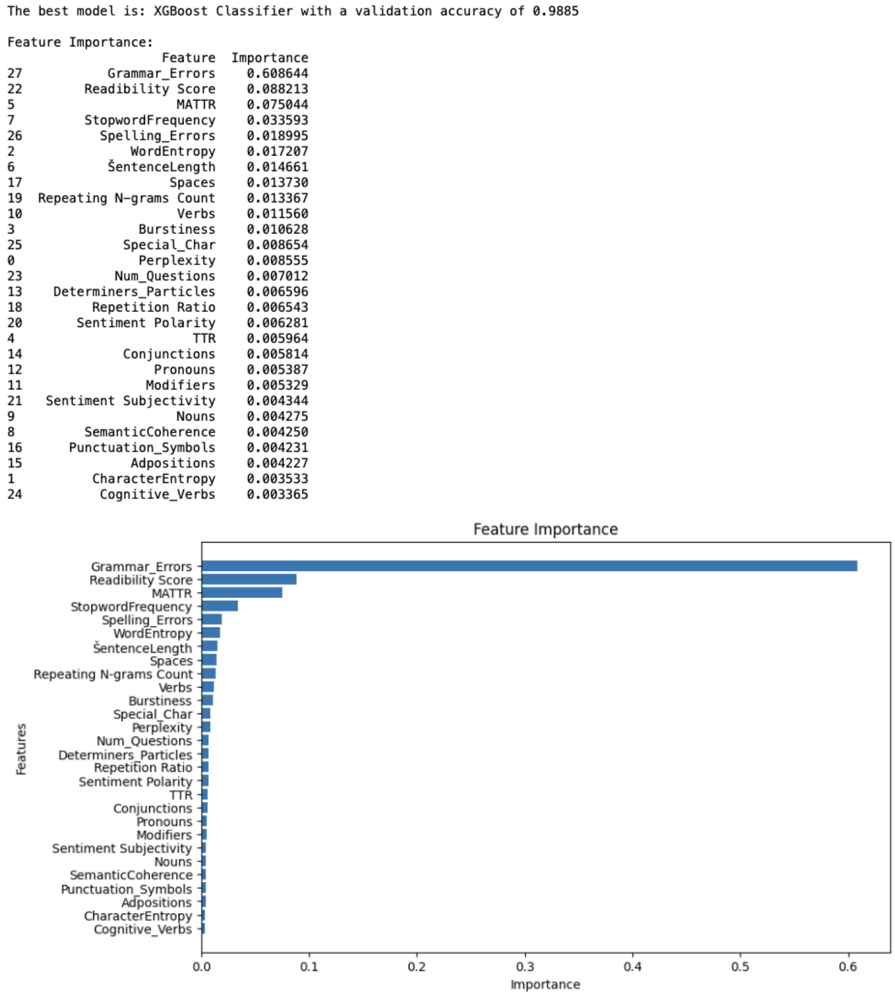
5.3 Dataset: LLM - Detect AI-Generated Text Dataset

The results in Table 3 demonstrate that XGBoost is the top-performing model, achieving an impressive validation accuracy and F1-score of 0.9885, and even good performance during testing, with an accuracy and F1-score of 0.9871. This highlights XGBoost's strong ability to generalize across different datasets. Other models, such as the Multilayer Perceptron (accuracy: 0.9832, F1-score: 0.9832) and SVM with Polynomial Kernel (accuracy: 0.9744, F1-score: 0.9745), also perform well but fall slightly short of XGBoost's performance.

In terms of feature importance (Figure 6), *Grammar Errors* once again stood out as the most critical feature for classification, with an importance score of 0.6084, significantly outweighing all other features. *Readability score* (0.0822) and *MATTR* (0.0754) follow as the next most influential features, indicating the model's reliance on both linguistic correctness and textual complexity. Additional features such as *stopword frequency, spelling errors, and word entropy* contribute meaningfully but with less importance. Less impactful features, including *punctuation symbols, cognitive verbs, and adpositions*, play minor roles in the classification process.

Classification Algorithms	**Accuracy**	**F1-score**
Validation		
Logistic Regression	0.9415	0.9415
K-Nearest Neighbor	0.9242	0.9243
SVM (Linear)	0.9602	0.9603
SVM (Polynomial)	0.9744	0.9745
SVM (Gaussian)	0.9650	0.9650
Naïve Bayes Classifier	0.8840	0.8847
Decision Tree	0.9689	0.9690
Random Forest	0.9538	0.9536
XGBoost	0.9885	0.9885
Multilayer Perceptron	0.9832	0.9832
Testing		
XGBoost	0.9871	0.9871

Table 3: Performance metrics for classification algorithms on dataset (LLM - Detect AI Generated Text)



```
The best model is: XGBoost Classifier with a validation accuracy of 0.9885

Feature Importance:
                    Feature  Importance
27           Grammar_Errors    0.608644
22        Readibility Score    0.088213
5                     MATTR    0.075044
7         StopwordFrequency    0.033593
26          Spelling_Errors    0.018995
2               WordEntropy    0.017207
6            ŠentenceLength    0.014661
17                   Spaces    0.013730
19  Repeating N-grams Count    0.013367
10                    Verbs    0.011560
3                Burstiness    0.010628
25             Special_Char    0.008654
0                Perplexity    0.008555
23            Num_Questions    0.007012
13      Determiners_Particles    0.006596
18         Repetition Ratio    0.006543
20       Sentiment Polarity    0.006281
4                       TTR    0.005964
14             Conjunctions    0.005814
12                 Pronouns    0.005387
11                Modifiers    0.005329
21   Sentiment Subjectivity    0.004344
9                     Nouns    0.004275
8         SemanticCoherence    0.004250
16      Punctuation_Symbols    0.004231
15              Adpositions    0.004227
1          CharacterEntropy    0.003533
24           Cognitive_Verbs    0.003365
```

Figure 6: Feature Importance - XGBoost Classifier

6 Testing the Website

The images (Figures 7,8 and 9) showcase a web application titled ”De-AI Cipher: Decoding the Language of Machines”, which I designed to analyze real-time input content for AI-generated or human-written text.

Here’s a detailed explanation of the different components and functionalities depicted in the interface:

1. The topmost section contains a text input box where users can enter or paste text. After entering the text, the user clicks the ”Analyze” button to process the text.

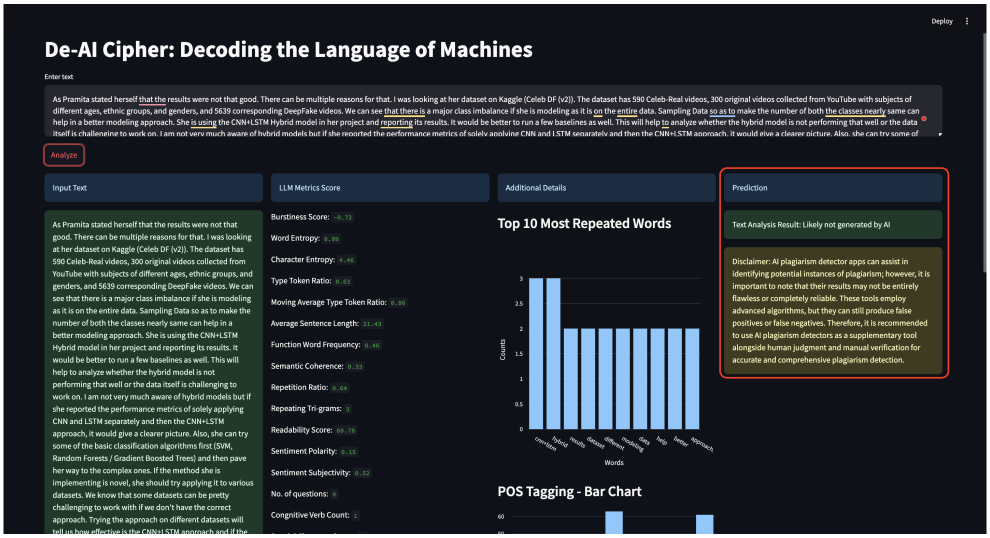
2. The LLM Metrics Score section provides the values of various linguistic and LLM metrics calculated for the respective input text.

3. The Additional Details section includes a chart titled ”Top 10 Most Repeated Words” visualizing the most frequently occurring words in the text, along with their counts. It also shows POS-tags distribution, top 5 most common Tri-grams and Bi-grams.

4. The application uses the calculated metrics to predict whether the text is AI-generated or human-written. The machine learning model used here the best one from the above 3 datasets.

A disclaimer is provided to clarify the limitations of AI-based plagiarism detection or text classification tools. These tools are not entirely flawless and might produce false positives or negatives. They should be used alongside human judgment for accurate and comprehensive detection.

The source code of this application is provided in the submission.



De-AI Cipher: Decoding the Language of Machines

Enter text

As Pramita stated herself that the results were not that good. There can be multiple reasons for that. I was looking at her dataset on Kaggle (Celeb DF (v2)). The dataset has 590 Celeb-Real videos, 300 original videos collected from YouTube with subjects of different ages, ethnic groups, and genders, and 5639 corresponding DeepFake videos. We can see that there is a major class imbalance if she is modeling as it is on the entire data. Sampling Data so as to make the number of both the classes nearly same can help in a better modeling approach. She is using the CNN+LSTM Hybrid model in her project and reporting its results. It would be better to run a few baselines as well. This will help to analyze whether the hybrid model is not performing that well or the data itself is challenging to work on. I am not very much aware of hybrid models but if she reported the performance metrics of solely applying CNN and LSTM separately and then the CNN+LSTM approach, it would give a clearer picture. Also, she can try some of

Input Text

As Pramita stated herself that the results were not that good. There can be multiple reasons for that. I was looking at her dataset on Kaggle (Celeb DF (v2)). The dataset has 590 Celeb-Real videos, 300 original videos collected from YouTube with subjects of different ages, ethnic groups, and genders, and 5639 corresponding DeepFake videos. We can see that there is a major class imbalance if she is modeling as it is on the entire data. Sampling Data so as to make the number of both the classes nearly same can help in a better modeling approach. She is using the CNN+LSTM Hybrid model in her project and reporting its results. It would be better to run a few baselines as well. This will help to analyze whether the hybrid model is not performing that well or the data itself is challenging to work on. I am not very much aware of hybrid models but if she reported the performance metrics of solely applying CNN and LSTM separately and then the CNN+LSTM approach, it would give a clearer picture. Also, she can try some of the basic classification algorithms first (SVM, Random Forests / Gradient Boosted Trees) and then pave her way to the complex ones. If the method she is implementing is novel, she should try applying it to various datasets. We know that some datasets can be pretty challenging to work with if we don't have the correct approach. Trying the approach on different datasets will tell us how effective is the CNN+LSTM approach and if the

LLM Metrics Score

Burstiness Score: -0.72

Word Entropy: 6.99

Character Entropy: 4.46

Type Token Ratio: 0.63

Moving Average Type Token Ratio: 0.86

Average Sentence Length: 21.43

Function Word Frequency: 0.46

Semantic Coherence: 0.33

Repetition Ratio: 0.64

Repeating Tri-grams: 3

Readability Score: 60.76

Sentiment Polarity: 0.15

Sentiment Subjectivity: 0.52

No. of questions: 0

Congnitive Verb Count: 1

Additional Details

Top 10 Most Repeated Words

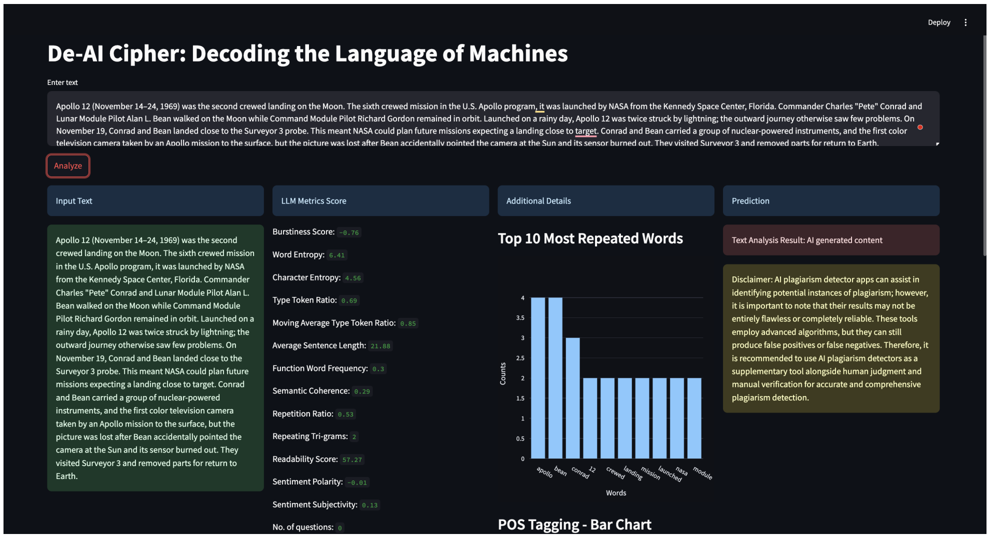
POS Tagging - Bar Chart

Prediction

Text Analysis Result: Likely not generated by AI

Disclaimer: AI plagiarism detector apps can assist in identifying potential instances of plagiarism; however, it is important to note that their results may not be entirely flawless or completely reliable. These tools employ advanced algorithms, but they can still produce false positives or false negatives. Therefore, it is recommended to use AI plagiarism detectors as a supplementary tool alongside human judgment and manual verification for accurate and comprehensive plagiarism detection.

Figure 7: The presentation review I wrote for Pramita is correctly classified as ”Likely not generated by AI”.



De-AI Cipher: Decoding the Language of Machines

Enter text

Apollo 12 (November 14–24, 1969) was the second crewed landing on the Moon. The sixth crewed mission in the U.S. Apollo program, it was launched by NASA from the Kennedy Space Center, Florida. Commander Charles "Pete" Conrad and Lunar Module Pilot Alan L. Bean walked on the Moon while Command Module Pilot Richard Gordon remained in orbit. Launched on a rainy day, Apollo 12 was twice struck by lightning; the outward journey otherwise saw few problems. On November 19, Conrad and Bean landed close to the Surveyor 3 probe. This meant NASA could plan future missions expecting a landing close to target. Conrad and Bean carried a group of nuclear-powered instruments, and the first color television camera taken by an Apollo mission to the surface, but the picture was lost after Bean accidentally pointed the camera at the Sun and its sensor burned out. They visited Surveyor 3 and removed parts for return to Earth.

Input Text

Apollo 12 (November 14–24, 1969) was the second crewed landing on the Moon. The sixth crewed mission in the U.S. Apollo program, it was launched by NASA from the Kennedy Space Center, Florida. Commander Charles "Pete" Conrad and Lunar Module Pilot Alan L. Bean walked on the Moon while Command Module Pilot Richard Gordon remained in orbit. Launched on a rainy day, Apollo 12 was twice struck by lightning; the outward journey otherwise saw few problems. On November 19, Conrad and Bean landed close to the Surveyor 3 probe. This meant NASA could plan future missions expecting a landing close to target. Conrad and Bean carried a group of nuclear-powered instruments, and the first color television camera taken by an Apollo mission to the surface, but the picture was lost after Bean accidentally pointed the camera at the Sun and its sensor burned out. They visited Surveyor 3 and removed parts for return to Earth.

LLM Metrics Score

Burstiness Score: `-0.76`

Word Entropy: `6.41`

Character Entropy: `4.56`

Type Token Ratio: `0.69`

Moving Average Type Token Ratio: `0.85`

Average Sentence Length: `21.88`

Function Word Frequency: `0.3`

Semantic Coherence: `0.29`

Repetition Ratio: `0.53`

Repeating Tri-grams: `2`

Readability Score: `57.27`

Sentiment Polarity: `-0.01`

Sentiment Subjectivity: `0.13`

No. of questions: `0`

Additional Details

Top 10 Most Repeated Words

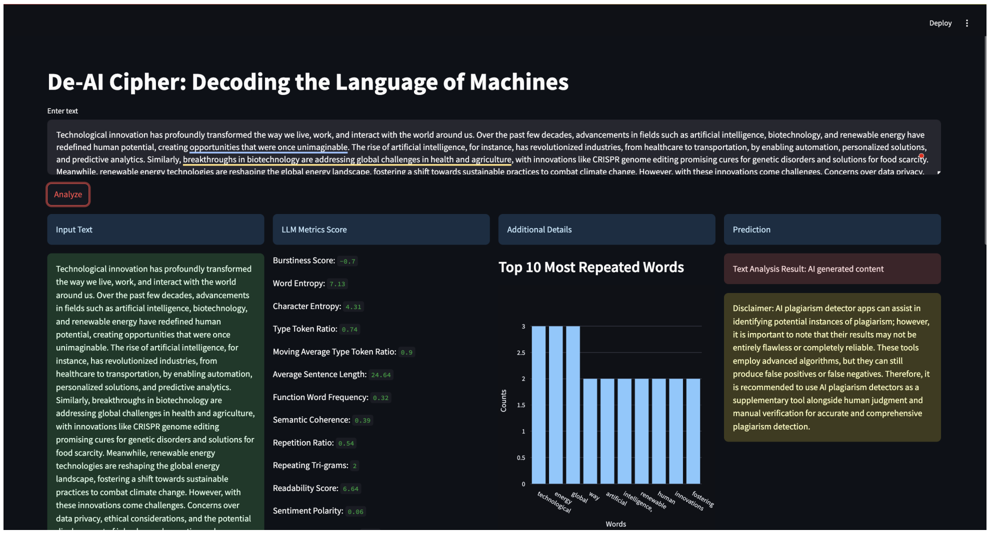
POS Tagging - Bar Chart

Prediction

Text Analysis Result: AI generated content

Disclaimer: AI plagiarism detector apps can assist in identifying potential instances of plagiarism; however, it is important to note that their results may not be entirely flawless or completely reliable. These tools employ advanced algorithms, but they can still produce false positives or false negatives. Therefore, it is recommended to use AI plagiarism detectors as a supplementary tool alongside human judgment and manual verification for accurate and comprehensive plagiarism detection.

Figure 8: The Wikipedia article is classified as ”Likely generated by AI” because it lacks grammatical errors



De-AI Cipher: Decoding the Language of Machines

Enter text

Technological innovation has profoundly transformed the way we live, work, and interact with the world around us. Over the past few decades, advancements in fields such as artificial intelligence, biotechnology, and renewable energy have redefined human potential, creating opportunities that were once unimaginable. The rise of artificial intelligence, for instance, has revolutionized industries, from healthcare to transportation, by enabling automation, personalized solutions, and predictive analytics. Similarly, breakthroughs in biotechnology are addressing global challenges in health and agriculture, with innovations like CRISPR genome editing promising cures for genetic disorders and solutions for food scarcity. Meanwhile, renewable energy technologies are reshaping the global energy landscape, fostering a shift towards sustainable practices to combat climate change. However, with these innovations come challenges. Concerns over data privacy,

Input Text

Technological innovation has profoundly transformed the way we live, work, and interact with the world around us. Over the past few decades, advancements in fields such as artificial intelligence, biotechnology, and renewable energy have redefined human potential, creating opportunities that were once unimaginable. The rise of artificial intelligence, for instance, has revolutionized industries, from healthcare to transportation, by enabling automation, personalized solutions, and predictive analytics. Similarly, breakthroughs in biotechnology are addressing global challenges in health and agriculture, with innovations like CRISPR genome editing promising cures for genetic disorders and solutions for food scarcity. Meanwhile, renewable energy technologies are reshaping the global energy landscape, fostering a shift towards sustainable practices to combat climate change. However, with these innovations come challenges. Concerns over data privacy, ethical considerations, and the potential

LLM Metrics Score

Burstiness Score: `-0.7`

Word Entropy: `7.13`

Character Entropy: `4.31`

Type Token Ratio: `0.74`

Moving Average Type Token Ratio: `0.9`

Average Sentence Length: `24.64`

Function Word Frequency: `0.32`

Semantic Coherence: `0.39`

Repetition Ratio: `0.54`

Repeating Tri-grams: `2`

Readability Score: `6.64`

Sentiment Polarity: `0.06`

Additional Details

Top 10 Most Repeated Words

Prediction

Text Analysis Result: AI generated content

Disclaimer: AI plagiarism detector apps can assist in identifying potential instances of plagiarism; however, it is important to note that their results may not be entirely flawless or completely reliable. These tools employ advanced algorithms, but they can still produce false positives or false negatives. Therefore, it is recommended to use AI plagiarism detectors as a supplementary tool alongside human judgment and manual verification for accurate and comprehensive plagiarism detection.

Figure 9: A 250-word paragraph generated from ChatGPT classified as ”Likely generated by AI”.

7 Challenges

1. Working with large language models (LLMs) requires significant computational resources, including high-performance GPUs. Training large models can take hours/-days, depending on the dataset size and model complexity, causing delays. The limited

availability of advanced hardware made it challenging for me to prepare the derived features and run BERT for baseline.

2. To achieve fair and robust performance, the dataset used for training or evaluation must accurately represent the diversity and generality of real-world scenarios. A biased or unrepresentative dataset can lead to models that fail to generalize across different use cases or domains. Example. Kaggle Datasets 1, 2, and 3 had grammatical errors as a major deciding factor when it comes for human-written text vs AI-generated text. In the real world, if a human is well-versed in the English language and makes no grammatical errors, his/her text will be marked as generated by AI.

3. Differentiating between AI-generated and human-generated text is an emerging problem with limited research. Therefore, it is challenging to develop benchmarks or metrics for reliably distinguishing them.

8 Future Scope

1. Enhance the analysis by using multiple variations of the Type-Token Ratio (TTR) to gain deeper insights into lexical diversity. For example: Root TTR, Corrected TTR etc. These variations provide complementary perspectives on lexical diversity, making the analysis more comprehensive and adaptable to different text types or lengths.

2. Expand the evaluation framework by incorporating additional readability metrics to capture the complexity of text from various angles. For example: Coleman-Liau Index, Automated Readability Index (ARI), SMOG Index (Simple Measure of Gobbledygook) etc. By using multiple algorithms, you can offer a more nuanced evaluation of text readability and adapt the analysis to different target audiences or domains.

3. Test the robustness, scalability, and generalizability of the methodology by applying it to a larger dataset. A larger dataset provides a ”big-picture” view of my methodology, revealing potential limitations, edge cases, or areas for improvement. Recently found dataset: Human vs. LLM Text Corpus consisting of 788922 unique records

9 Acknowledgment

I would like to extend my sincere thanks to all those who played a crucial role in the successful completion of this project. Special appreciation goes to Professor Pei Wang for his invaluable guidance and insights.

References

[1] André, C. M., Eriksen, H. F., Jakobsen, E. J., Mingolla, L. C., Thomsen, N. B. (2023). Detecting AI Authorship: Analyzing Descriptive Features for AI Detection.

[2] Fraser, K. C., Dawkins, H., Kiritchenko, S. (2024). Detecting ai-generated text: Factors influencing detectability with current methods. arXiv preprint arXiv:2406.15583.

[3] Mukherjee, S. (2023, June 22). Exploring Burstiness: Evaluating Language Dynamics in LLM-Generated Texts. Medium. https://ramblersm.medium.com/exploring-burstiness-evaluating-language-dynamics-in-llm-generated-texts-8439204c75c1

[4] Depala, R. (2023, August 31). Type Token Ratio in NLP. Medium. https://medium.com/@rajeswaridepala/empirical-laws-ttr-cc9f826d304d

[5] Reviriego, P., Conde, J., Merino-Gómez, E., Martínez, G., Hernández, J. A. (2023). Playing with words: Comparing the vocabulary and lexical richness of ChatGPT and humans. arXiv preprint arXiv:2308.07462.

[6] Herbold, S., Hautli-Janisz, A., Heuer, U. et al. A large-scale comparison of human-written versus ChatGPT-generated essays. Sci Rep 13, 18617 (2023). https://doi.org/10.1038/s41598-023-45644-9

[7] Heikkilä, M. (2022, December 19). How to spot AI-generated text. MIT Technology Review. https://www.technologyreview.com/2022/12/19/1065596/how-to-spot-ai-generated-text/

[8] Ma, Y., Liu, J., Yi, F., Cheng, Q., Huang, Y., Lu, W., Liu, X. (2023). AI vs. Human–Differentiation Analysis of Scientific Content Generation. arXiv preprint arXiv:2301.10416.

[9] Fröhling, L., Zubiaga, A. (2021). Feature-based detection of automated language models: tackling GPT-2, GPT-3 and Grover. PeerJ Computer Science, 7, e443.

[10] Zheng, W. (2024). AI vs. Human: A Comparative Study of Cohesion and Coherence in Academic Texts between Human-Written and ChatGPT-Generated Texts.

[11] Rad, M. H., Farsi, F., Bali, S., Etezadi, R., Shamsfard, M. (2024). RFBES at SemEval-2024 Task 8: Investigating Syntactic and Semantic Features for Distinguishing AI-Generated and Human-Written Texts. arXiv preprint arXiv:2402.14838.

[12] Morales-Márquez, L. E., González, E. B., Avendaño, D. E. P. (2023). Artificial Intelligence-Based Text Classification: Separating Human Writing from Computer Generated Writing. In IberLEF@ SEPLN.

[13] , ., , ., , . (2023). Linguistic Analysis of Human-and AI-Created Content in Academic Discourse. . , (10), 47-67.

[14] Heikkilä, M. (2022, December 19). How to spot AI-generated text. MIT Technology Review. https://www.technologyreview.com/2022/12/19/1065596/how-to-spot-ai-generated-text/

[15] Steere, E. (2024). Ways to distinguish AI-composed essays from human-composed ones (opinion). Inside Higher Ed — Higher Education News, Events and Jobs. https://www.insidehighered.com/opinion/career-advice/teaching/2024/07/02/ways-distinguish-ai-composed-essays-human-composed-ones

[16] Robinson, R. (2024, November 8). Uncovering Repetition: How Syntactic Templates Reveal Patterns in AI-Generated Text. ComplexDiscovery. https://complexdiscovery.com/uncovering-repetition-how-syntactic-templates-reveal-patterns-in-ai-generated-text/

[17] Monje, S., Ulene, S., Gimovsky, A. C. (2024). Identifying ChatGPT-written Patient Education Materials Using Text Analysis and Readability. American Journal of Perinatology.

[18] Opara, C. (2024, July). StyloAI: Distinguishing AI-Generated Content with Stylometric Analysis. In International Conference on Artificial Intelligence in Education (pp. 105-114). Cham: Springer Nature Switzerland.

[19] Schaaff, K., Schlippe, T., Mindner, L. (2023). Classification of Human-and AI-Generated Texts for English, French, German, and Spanish. arXiv preprint arXiv:2312.04882.

[20] AI and Academic Writing: Humans are Still Needed — BridgeText - Online Dissertation Writing Service and Help. (2023). BridgeText - Online Dissertation Writing Service and Help. https://www.bridgetext.com/ai-and-academic-writing-humans-are-still-needed

[21] How To Identify AI-Generated Text? – Originality.AI. (n.d.). Originality.ai. https://originality.ai/blog/identify-ai-generated-text

[22] Thomas Hirschmann. (2022, December 8). There are several key differences between text written by a human and text generated by a machine learning model like ChatGPT. Here are three of the most important ones: Style: Human-written text tends to have a more natural and varied style, with a wide range of sentence structures and word choices. Linkedin.com. https://www.linkedin.com/pulse/chatgpt-beware-how-spot-ai-generated-text-thomas-hirschmann/

[23] Wang, P., Li, L., Ren, K., Jiang, B., Zhang, D., Qiu, X. (2023). SeqXGPT: Sentence-level AI-generated text detection. arXiv preprint arXiv:2310.08903.

[24] Gaggar, R., Bhagchandani, A., Oza, H. (2023). Machine-generated text detection using deep learning. arXiv preprint arXiv:2311.15425.