Mirror Minds : An Empirical Study on Detecting LLM-Generated Text via LLMs

Josh Baradia* PES University Karnataka, India joshbaradia22@gmail.com

Shubham Gupta* IIT Jodhpur Rajasthan, India gupta.37@iitj.ac.in

Suman Kundu IIT Jodhpur Rajasthan, India suman@iitj.ac.in

Abstract

The use of large language models (LLMs) is inevitable in text generation. LLMs are intelligent and slowly replacing the search engines. LLMs became the de facto choice for conversation, knowledge extraction, and brain storming. This study focuses on a question: 'Can we utilize the generative capabilities of LLMs to detect AI-generated content?' We present a methodology and empirical results on four publicly available data sets. The result shows, with 90% accuracy it is possible to detect AI-generated content by a zero-shot detector utilizing multiple LLMs. [1]

1 Introduction

The rapid advancement of large language models (LLMs) has elevated their text generation capabilities to levels comparable to human writing (OpenAI, 2024; Anthropic, 2023; Chowdhery et al., 2023). LLMs have become integral to various aspects of daily life and are increasingly pivotal in numerous professional workflows (Veselovsky et al., 2023). They aid in tasks such as creating advertising slogans (Murakami et al., 2023), composing news articles (Yanagi et al., 2020), and generating stories (Yuan et al., 2022). Additionally, the influence of LLMs is profoundly shaping the development of many sectors and disciplines, including education (Susnjak, 2022), law (Cui et al., 2024), and medicine (Thirunavukarasu et al., 2023). However, the remarkable proficiency of generative language models in producing text has simultaneously heightened worries about their potential misuse in fields such as phishing, spreading misinformation, and academic dishonesty. Regrettably, humans are only marginally better than chance at distinguishing between AI-generated text and text written by people (Gehrmann et al., 2019a). Consequently, we aim to create an automated system that can accurately identify AI-generated texts to help prevent their harmful applications.

LLMs no longer require task-specific learning or alignment between task inputs and desired outputs because they have already acquired most necessary knowledge during pre-training. Instruction tuning helps to better align the model with the anticipated responses for user tasks. Motivated by this, we do an empirical study on a question that "**Can a LLM detect the content written by AI?**".

We developed a three-step methodology to identify AI-generated texts. In the first step, we generate a contextually relevant query from the input through large language model. Essentially, we aim to create an input prompt that logically follows the given text. The process of generating this prompt is very fast ($\sim$2 seconds) due to the nature of zero-shot inference in LLMs. Once the prompt is generated, we pass it to two different large language models, each of which produces a response of approximately the same length as the input text. These generated texts are then compared with original input for the final classification. We use BLEU (BiLingual Evaluation Understudy) (Papineni et al., 2002) and Meteor (Banerjee and Lavie, 2005) score as comparative measures. We tested our methodology on four publicly available datasets, and the results show that our pipeline can detect AI-generated text with an accuracy of approximately 90%.

2 Related Work

Earlier detection methods primarily relied on feature-based approaches, such as analyzing the frequency of rare bigrams (Grechnikov et al., 2009), n-gram frequencies (Badaskar et al., 2008), or top-k word patterns as used in GLTR (Gehrmann et al., 2019b). However, as machine-generated text becomes increasingly sophisticated, a shift toward

*Both authors contribute equally.

[1]Code repository: https://github.com/shubhamgpt007/MirrorMinds-LLMDetector

Proceedings of the 1st Workshop on GenAI Content Detection (GenAIDetect), pages 59–67
January 19, 2025.

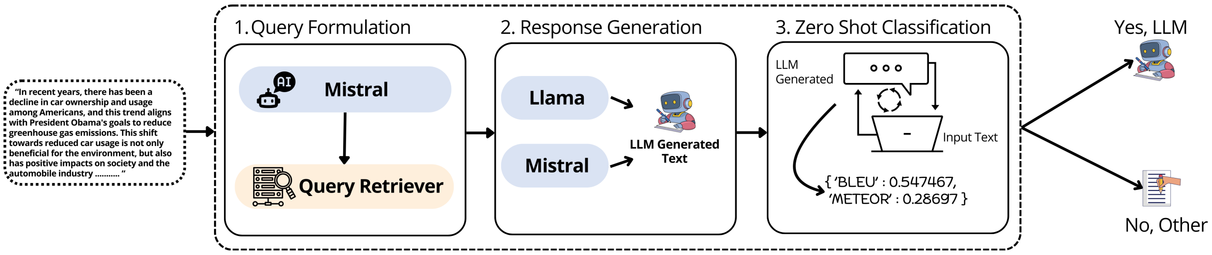

Figure 1: An overview of our proposed approach to detect LLM generated text

trained-based methods has emerged. For exmaple, recent literature to detect AI-generated texts include Guo et al. (2023), who fine-tuned RoBERTa to identify texts in English and Chinese as either AI-generated or human-written, facing challenges with overfitting that degraded performance on out-of-domain data. Also, OpenAI Text Classifier (OpenAI, Jan 2023) and GPTZero (Tian, 2023) that detects AI-generated text by training on the input data. A notable challenge with these models is the need for periodic retraining to keep pace with updates and advancements in new large language models (LLMs). Another approach, DetectGPT by (Mitchell et al., 2023), employs a zero-shot classifier that assumes AI-generated texts exhibit lower model probabilities compared to originals, but struggles with short texts due to its need for lengthy inputs to understand context. Additionally, Kirchenbauer et al. (2023) explored using watermarks in AI-generated texts to aid detection, though this method can compromise text readability and watermark removal presents further difficulties. In a very recent times, DNA-GPT (Yang et al., 2024) proposed which provides a distinctive and effective approach to identifying GPT-generated text by leveraging the inherent differences in continuation patterns between human-written and AI-generated content.

3 Methodology

3.1 Overview of LLM-Detector

Figure 1 illustrates the structure of our proposed zero-shot classification methodology. This method employs a large language model-based detector for zero-shot inferencing to categorize input text as either AI-generated or not. Our zero-shot classification model comprises three modules: 1) Query Formulation, which identifies the contextual query from the input text using a large language model; 2) Response Generation, which creates multiple responses, approximately equal in length to the input text, for the identified query using various large language models; 3) Classification Module, which determines whether the input text is AI-generated or not. The subsequent sections provide a detailed description of each module.

3.2 Query Formulation

The Query Formulation module is a crucial component of our zero-shot classification methodology, designed to interpret and extract the contextual query from the input text. Utilizing a large language model, this module analyzes the text to identify key themes, questions, or topics inherent within the content. Importance of query formulation is in ensuring precise interpretation of complex inputs, aligning subsequent analysis with the thematic intent of the original text. For this, we have used Mistral-7B (Jiang et al., 2023) model that is capable to handle complex queries and generate responses that are contextually relevant and linguistically accurate. The effectiveness of this process is paramount as it sets the foundation for subsequent modules. By distilling complex and varied input texts into precise queries, this module ensures that the response generation process is accurately aligned with the thematic direction of the initial input, thereby facilitating a more targeted and coherent analysis in later stages of classification.

Given a textual input denoted as $T = \{w_1, w_2, \cdots, w_n\}$, consisting of a sequence of n tokens, were tokenized and fed into the Mistral LLM model to obtain the query text Q as follows:

$$Q = \text{Mistral}(T) \tag{1}$$

LLM's Input Prompt:
Generate a query that encapsulates the main theme of the following text.{text}

3.3 Response Generation

Response Generation module leverages multiple large language models to produce responses that are contextually aligned and approximately equal in length to the original input text from the identified query Q. We have used two LLMs viz. Llama3-7B (Dubey et al., 2024) and Mistral-7B (Jiang et al., 2023) to generate the response with the extracted query prompt. Given a query Q, generate the responses from the different LLMs as follows:

$$T' = \text{Mistral}(Q); \;\; T'' = \text{Llama}(Q) \qquad (2)$$

LLM's Input Prompt:
Generate a response between {textWordCount} to {1.1 *textWordCount} words for the following question.{Q}

3.4 Classification Module

The similarity of the generated text (T' and T'') is assessed by comparing it to the reference input text T. We employ two well-known metrics for this comparison: the BLEU (Papineni et al., 2002) and Meteor (Banerjee and Lavie, 2005) scores, which help quantify the closeness of the generated text to the original text in terms of syntax and semantics. For each LLM, a BLEU and a Meteor score are calculated independently using generated and original text. After obtaining these scores, we select max score among different generated texts BLEU and Meteor score to get a final BLEU score (B_f) and a final Meteor score (M_f). Finally, by setting a specific threshold for these scores, we can determine whether the input text resembles AI-generated content or not, aiding in distinguishing between the two.

$$B_f = max(BLEU(T, T'), BLEU(T, T'')) \quad (3)$$

$$M_f = max(Meteor(T, T'), Meteor(T, T'')) \quad (4)$$

4 Experiments and Results

4.1 Datasets & Evaluation Metrics

As our objective is to evaluate model performance on the task to identify AI-generated text, we chose four publicly available datasets Daigt Data (Kleczek, 2013) which consists essay generated from the Falcon-180B (DS1) and Llama-70B (DS2), Palm dataset (Muhammad, 2023) (DS3), and Human-and-AI dataset (Shayan, 2023) (DS4). Human-and-AI dataset originally consists 432k+ entries but for this experiment we have filtered 10K human and 10k AI-generated essays. The details of these datasets can be found in Table 1.

In order to evaluate the performance of our methodology, we employed the BLEU and Meteor scores to assess the similarity between the generated responses and the original text. The average scores for each metric across dataset are reported in the results section. Additionally, we calculated accuracy based on thresholds set for these scores specifically, a BLEU score of 0.52 and a Meteor score of 0.27. These values are found using grid search with varying BLEU from 0.5 to 0.6 and Meteor from 0.2 to 0.3. To validate this combination, we tested various BLEU and Meteor score configurations using the DS4 dataset, which contains both human-generated and LLM-generated text. As shown in Figure 2, the optimal generalization results were achieved with a BLEU score of 0.52 and a Meteor score of 0.27.

Table 1: Distribution of data across different datasets.

Dataset	Description	Distribution
DS1	DAIGT dataset generated from Falcon-180B	1055 rows
DS2	DAIGT dataset generated from Llama-70B and Falcon-180B	7000 rows
DS3	Essay dataset generated by PaLM	1384 rows
DS4	Articles dataset consists of Human and AI generated text	20000 rows

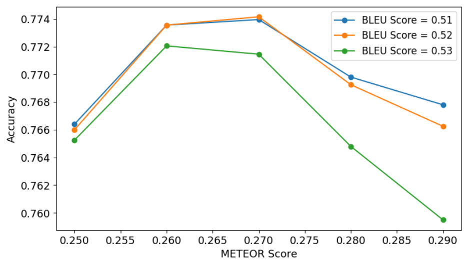

Figure 2: Comparison of accuracy among various BLEU and Meteor scores for DS4 dataset.

4.2 Experimental Setup

We configured Llama and Mistral LLM's using parameters viz. num_return_sequences = 1, top_p = 0.95, top_k=40, num_beams=2, max_length = 1.1*textWordCount. These models are run on NVIDIA A30 with 24 GB of GPU memory. As part of textual pre-processing, symbols such as '@', '#' and hyperlinks are removed.

5 Results

Table 2 shows the performance of our detection system under various LLM configurations. We utilized the Mistral LLM for query extraction and employed two LLMs, Llama3 and Mistral, for response generation. The table compares the outcomes across three models: the first model uses Llama3 for response generation, the second employs Mistral, and the third integrates both LLMs in the response generation process. We have experimented our methodology across all the datasets mentioned in the Table 1. In the results table, we have reported BLEU, Meteor Score and Accuracy of each model.

The results presented in Table 2 show that using a single LLM for response generation yields specific scores: the Llama model achieved a BLEU score of 0.61 and a Meteor score of 0.33, while the Mistral model scored 0.56 and 0.31 for the DS1, respectively. However, combining both LLMs led to improved scores of 0.62 for BLEU and 0.34 for Meteor. Furthermore, the accuracy of the combined LLM methodology was notably higher, at 0.92, outperforming the single-model setups. In a similar manner, we can observe the results for DS2 and DS3 datasets.

Table 2: Performance comparison of different models across datasets

Dataset	Model	BLEU	METEOR	Accuracy
DS1	Ours-Llama	0.61	0.33	0.90
	Ours-Mistral	0.56	0.31	0.63
	Ours-(Llama+Mistral)	**0.62**	**0.34**	**0.92**
DS2	Ours-Llama	0.61	0.31	0.83
	Ours-Mistral	0.56	0.31	0.49
	Ours-(Llama+Mistral)	**0.62**	**0.32**	**0.86**
DS3	Ours-Llama	0.62	0.33	0.96
	Ours-Mistral	0.59	0.32	0.90
	Ours-(Llama+Mistral)	**0.63**	**0.34**	**0.98**
DS4	Ours-Llama	0.59	0.31	0.77
	Ours-Mistral	0.58	0.31	0.69
	Ours-(Llama+Mistral)	**0.60**	**0.32**	**0.78**

For the DS4 datasets, which include both human-written and AI-generated text, we observed a consistent pattern in the performance metrics. When using single LLMs, the Llama model achieved a BLEU score of 0.59 and a Meteor score of 0.31, while the Mistral model scored 0.58 for BLEU and 0.31 for Meteor. However, the combined LLM model showed improved results, with a BLEU score of 0.60 and a Meteor score of 0.32. In terms of accuracy, the final model with both LLMs significantly outperformed the single-model setups.

The figure above shows how accuracy changes with METEOR scores for various BLEU levels (0.51, 0.52, and 0.53). Across all BLEU scores, accuracy peaks at a METEOR value of 0.27, indicating that this is the best METEOR range for maximum accuracy. BLEU ratings of 0.51 and 0.52 show comparable patterns, with somewhat better accuracy at 0.52, especially near the peak. However, BLEU score 0.53 has consistently worse accuracy throughout all METEOR scores, with a greater fall after the peak. This implies that, while higher BLEU scores might increase performance up to a certain point, an excessively high BLEU score may have a declining or negative influence on accuracy, emphasising the need of balancing METEOR and BLEU scores to obtain optimal performance.

6 Usability Analysis

6.1 Robustness on different word ranges

In our study, we evaluated the performance of our zero-shot detector across different text lengths by selecting samples with 0-100, 100-200, 200-300, 300-400, 500-600, and over 600 words. The findings, presented in Table 3, show a distinct pattern. The detector struggles with accuracy in smaller samples (0-100 words) but shows marked improvements as the word count increases. This increase in performance in larger text samples suggests that the zero-shot detector benefits from more contextual information, which may be lacking in shorter texts. This behavior could also be influenced by the domain-specific characteristics or statistical properties of the LLM models. We have discussed case studies related to short-length, long-length, and human-written text in Appendix section 9 .

Table 3: Comparison of datasets across different word ranges.

Dataset	Metric	0-100	100-200	200-300	300-400	400-500	500-600	>600
DS1	accuracy	0.50	0.82	0.90	0.96	0.89	-	-
	# rows	2	51	515	432	55	-	-
DS2	accuracy	0.0	0.41	0.70	0.83	0.91	0.90	0.89
	# rows	1	27	714	2161	2569	1208	320
DS3	accuracy	-	1.0	0.98	0.99	0.97	0.98	0.72
	# rows	-	1	180	658	474	60	11
DS4	accuracy	0.04	0.65	0.75	0.76	0.85	0.83	0.87
	# rows	84	1586	4790	6159	3790	1676	1915

6.2 Inference time for the detection

In our detector, we have two core phases Query Formulation and Response Generation which takes time for the final classification. In Table 4, we have reported the query formulation time (QFT) and response generation time (RGT) phases across all the datasets. From the value, it is evident that Query Formulation Time (QFT) is relatively stable across datasets, ranging from 1.15 seconds to 1.95 seconds, suggesting that the extraction process is largely unaffected by dataset complexity. In ad-

Table 4: Analysis of time (in sec) taken across different phases 1) Query Formulation Time (QFT) and 2) Response generation Time (RGT)in zero-shot detector.

Dataset	QET	RGT	
		Llama3	Mistral
DS1	1.95	13.38	10.03
DS2	1.15	19.83	15.63
DS3	1.18	17.64	13.48
DS4	1.81	17.11	12.72

dition, Response Generation Time (RGT) varied more significantly, with Llama3 showing longer processing times across all datasets compared to Mistral. Specifically, both models took the longest time on DS2, indicating higher complexity or computational demands associated with this dataset.

7 Conclusion

While Large Language Models (LLMs) have demonstrated impressive capabilities in generative tasks, mitigating their potential misuse remains crucial. This paper provides an empirical study by introducing a simple yet effective method for detecting AI-generated text. Our zero-shot detector leverages the extensive knowledge acquired by LLMs during pre-training, enabling them to identify their own generated outputs. Note that this methodology does not involve any training for classification; instead, it classifies input text using zero-shot inference. The process begins by extracting the main context of the input text, using the LLM to generate a corresponding response. This approach facilitates the detection of AI-generated text by comparing the similarity between the input and the generated text. Instruction tuning refines the model's alignment with the user expected responses in text detection tasks. We evaluated our method across four publicly available datasets, which cover responses generated by different LLMs and including both in-domain and out-of-domain (OOD) data. The results confirm that our approach effectively detects texts produced by LLMs. Future efforts will focus on enhancing detection capabilities for short texts and extending the application to other areas such as fake news detection.

8 Limitations

Our study has some limitations, primarily because it relies exclusively on two large language models, Mistral and Llama3. However, numerous state-of-the-art LLMs cited in the literature may outperform these and address challenges with short text detection more effectively. Additionally, while we utilize standard evaluation metrics like BLEU and Meteor, other metrics such as Perplexity and ROUGE might provide a more accurate assessment of machine translation quality.

Acknowledgement

Suman Kundu is supported by MeitY, Government of India under the grant no.4(2)/2024-ITEA.

References

Anthropic. 2023. Model card and evaluations for claude models. Accessed on 16.09.2024.

Sameer Badaskar, Sachin Agarwal, and Shilpa Arora. 2008. Identifying real or fake articles: Towards better language modeling. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113.

Jiaxi Cui, Munan Ning, Zongjian Li, Bohua Chen, Yang Yan, Hao Li, Bin Ling, Yonghong Tian, and Li Yuan. 2024. Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model. *Preprint*, arXiv:2306.16092.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019a. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019b. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

EA Grechnikov, GG Gusev, AA Kustarev, and A Raigorodsky. 2009. Detection of artificial texts. *RCDL2009 Proceedings. Petrozavodsk*, pages 306–308.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *Preprint*, arXiv:2301.07597.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17061–17084. PMLR.

D. Kleczek. 2013. Daigt proper train dataset. Accessed on 16.09.2024.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

R. Muhammad. 2023. Palm from google gen-ai. Accessed on 16.09.2024.

Soichiro Murakami, Sho Hoshino, and Peinan Zhang. 2023. Natural language generation for advertising: A survey. *Preprint*, arXiv:2306.12719.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. Jan 2023. Ai text classifier. Accessed on 16.09.2024.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

G. Shayan. 2023. Ai vs human text. Accessed on 16.09.2024.

Teo Susnjak. 2022. Chatgpt: The end of online exam integrity? *Preprint*, arXiv:2212.09292.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature medicine*, 29(8):1930–1940.

Edward Tian. 2023. Gptzero: An ai text detector. Accessed on 16.09.2024.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023. Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks. *ArXiv*, abs/2306.07899.

Yuta Yanagi, Ryohei Orihara, Yuichi Sei, Yasuyuki Tahara, and Akihiko Ohsuga. 2020. Fake news detection with generated comments for news articles. In *2020 IEEE 24th International Conference on Intelligent Engineering Systems (INES)*, pages 85–90.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Ruth Petzold, William Yang Wang, and Haifeng Chen. 2024. DNA-GPT: Divergent n-gram analysis for training-free detection of GPT-generated text. In *The Twelfth International Conference on Learning Representations*.

Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story writing with large language models. IUI '22, page 841–852, New York, NY, USA. Association for Computing Machinery.

9 Appendix

9.1 Example of short length text

We analyzed a sample from the DS4 dataset, originally labeled as AI-generated. When testing our zero-shot detector on this sample, it has BLEU scores of 0.37 and 0.26, and METEOR scores of 0.19 and 0.16 for Llama3 and Mistral, respectively. The quantitative data clearly indicates that our method struggles to recognize short-length LLM-generated texts. Additionally, it is evident that the context of the generated text does not align with the context of the input text.

Input Text:
Creating a new substance from chemicals is a complex process that requires extensive research. First, one must understand the chemical composition of different substances and learn the specific names of chemical compounds. Once knowledge is attained, experiments can be conducted in a lab to test different chemical reactions. The benefits of trial and error far outweigh the risks, and the potential for creating something incredibly valuable is infinite.

Query:
What are the steps involved in creating a new substance from chemicals and why is it a valuable process?

Llama Text:
The process of creating a new substance from chemicals involves several steps. First, chemists identify the desired properties of the new substance, such as its melting point, boiling point, and solubility. Next, they select the appropriate chemicals and react them.

Mistral Text:
Creating a new substance from chemicals, also known as organic synthesis or chemical synthesis, is a valuable process due to its potential to yield novel compounds with unique properties. The steps involved include: 1. Identifying the target.

9.2 Example of long length text

We analyzed a sample from the DS1 dataset, originally labeled as AI-generated. When testing our zero-shot detector on this sample, it has BLEU scores of 0.59 and 0.53, and METEOR scores of 0.33 and 0.28 for Llama3 and Mistral, respectively. The quantitative data clearly indicates that our method identifies this LLM-generated text as correct. We can also observe the generated text that it is contextually aligned and have similar patterns with input text.

Input Text:
When people ask for advice, they sometimes talk to more than one person. This is because seeking multiple opinions can help someone make a better choice. For example, if someone wants to buy a new phone, they might ask their friends, family, and even do some research online. By doing this, they can get different perspectives on which phone to buy, the pros and cons of each phone, and even the best deals available. Seeking multiple opinions can also help someone avoid making a bad decision. For instance, if someone is considering dropping out of school, they might talk to their parents, teachers, and even a counselor. By doing this, they can get advice on the consequences of dropping out, the benefits of staying in school, and even alternative options that they might not have considered before. In conclusion, seeking multiple opinions can help someone make a better choice by providing them with different perspectives, information, and advice. It can also help them avoid making a bad decision by considering all the options available. Therefore, it is always a good idea to seek multiple opinions before making a big decision.

Query:
What are the benefits of seeking multiple opinions before making a decision?

Llama Text:
Seeking multiple opinions before making a decision can have numerous benefits. Firstly, it allows individuals to gain a more comprehensive understanding of the situation and the potential outcomes. By hearing from different perspectives, individuals can identify potential risks and challenges that they may not have considered otherwise. This can help them make a more informed decision that takes into account a wider range of factors. Another benefit of seeking multiple opinions is that it can help individuals avoid making impulsive decisions. When faced with a decision, it's easy to get caught up in the moment and make a choice without fully considering the consequences. By taking the time to seek out multiple opinions, individuals can slow down the decision-making process and make a more thoughtful choice. Additionally, seeking multiple opinions can help individuals build stronger relationships with others. When individuals take the time to listen to and consider the opinions of others, they demonstrate that they value and respect those individuals. This can help build trust and strengthen relationships over time. Finally, seeking multiple opinions can help individuals develop a greater sense of confidence in their decision-making abilities. When individuals take the time to consider multiple perspectives.

Mistral Text:
Seeking multiple opinions before making a decision is an essential practice that can bring numerous benefits. Here are some reasons why it's a good idea to gather various perspectives before finalizing a choice: 1. Broaden your understanding: Each person brings a unique perspective based on their experiences, knowledge, and expertise. By seeking multiple opinions, you can broaden your understanding of the issue at hand and gain new insights that you may not have considered otherwise. 2. Identify potential biases: When making a decision, it's essential to be aware of any potential biases that may influence your thinking. Seeking multiple opinions can help you identify any biases that you may not have been aware of and allow you to make a more objective decision. 3. Improve decision-making quality: Multiple opinions can help improve the quality of your decision-making. By considering various perspectives, you can weigh the pros and cons more thoroughly and make a more informed decision. 4. Enhance problem-solving skills: Seeking multiple opinions.

9.3 Example of human-written text

We analyzed a sample from the DS4 dataset, originally labeled as human-written. When testing our zero-shot detector on this sample, it has BLEU scores of 0.41 and 0.29, and METEOR scores of 0.21 and 0.17 for Llama3 and Mistral, respectively. We can also see that while LLMs use complex vocabulary to respond to queries, humans tend to narrate using simpler words.

Input Text:
The author that evaluates how well the author supports the idea that studying Venus is a worth pursuit despite the danger. By saying" NASA is working on other approaches to studying Venus Food example some simplified elections made of silicon carbide have been tested in a chamber simulating". This evidence support my claim because the people AOE studying Food big things like how well studying Venus is a worthy pursuit. Another claim that evaluates how well the author supports the idea that studying Venus is a worth pursuit despite. In paragraph 8 it says" striving to meet the challenge presented by Venus has value. Not only because of the insight to be gained on the planet itself. Which explain that studying Venus is a worth pursuit despite. The conclusion I have about" the challenge of exploring Venus" was. It seems like if u study it itś a good thing. Because its very important to NASA it says that in the Leading. So maybe if u do it you would be successful.

Query:
What are the reasons why studying Venus is a worth pursuit despite the challenges and dangers it presents?

Llama Text:
Studying Venus is a worthwhile pursuit despite the challenges and dangers it presents because of the significant scientific and technological advancements it can bring about. Venus is often referred to as Earthś "sister planet" due to its similar size and proximity to the sun, making it an ideal target for studying planetary formation and evolution. By studying Venus, scientists can gain valuable insights into the early history of our solar system and the processes that shaped the planets. Furthermore, Venus offers a unique opportunity to study a planet with a thick atmosphere, which can provide valuable insights into the Earthś own atmospheric processes. The extreme conditions on Venus, such as the crushing pressure and scorching temperatures, also present a challenge for scientists to develop new technologies and instruments that can withstand such conditions. This can lead to breakthroughs in fields such as materials science and engineering. Additionally, studying Venus can also provide valuable information about the potential for life on other planets.

Mistral Text:
Studying Venus, the second planet from the Sun, presents unique challenges and dangers due to its extreme conditions. However, these challenges make Venus a worthwhile pursuit for scientists and explorers for several reasons. Firstly, Venus is the closest planet to Earth, making it an accessible target for exploration. Despite its inhospitable environment, understanding Venus's geology, atmosphere, and climate can provide valuable insights into the early history of Earth and the solar system as a whole. For instance, Venus's volcanic activity and corrosive atmosphere offer clues to the conditions that shaped our own planet billions of years ago. Secondly, Venus is the only planet in our solar system with a similar size, mass, and composition to Earth. Studying Venus can help us better understand the processes that shape terrestrial planets.