Exploring the Interpretability of AI-Generated Response Detection with Probing

Ikkyu Choi
ETS, Princeton, NJ.
ichoi001@ets.org

Jiyun Zu
ETS, Princeton, NJ.
jzu@ets.org

Abstract

Multiple strategies for AI-generated response detection have been proposed, with many high-performing ones built on language models. However, the decision-making processes of these detectors remain largely opaque. We addressed this knowledge gap by fine-tuning a language model for the detection task and applying probing techniques using adversarial examples. Our adversarial probing analysis revealed that the fine-tuned model relied heavily on a narrow set of lexical cues in making the classification decision. These findings underscore the importance of interpretability in AI-generated response detectors and highlight the value of adversarial probing as a tool for exploring model interpretability.

1 Introduction

Modern foundation language models have demonstrated the ability to generate coherent, well-structured text across a wide range of domains (Li et al., 2024; Zhao et al., 2023). This capability has affected various aspects of writing, including assignments and assessments that require learners to write original responses. As a result, educators and assessment professionals have become increasingly interested in distinguishing between human-written and AI-generated responses (Jiang et al., 2024). This task, which we refer to as AI-generated response detection in this paper, is the focus of our study.

The growing interest in, and demand for, AI-generated response detection has led to the development of algorithmic detectors, many of which are themselves based on language models. Some utilize the in-context learning capability of these models combined with prompt engineering, while others employ supervised fine-tuning on custom datasets designed for the detection task (see, e.g., Fraser et al., 2025 and Wu et al., 2025). These detectors are often marketed as highly accurate, with reported classification accuracy routinely exceeding 90 percent. However, similar to other inferences made by language models, the decisions from these detectors are opaque and difficult to interpret. This lack of transparency is particularly concerning in AI-generated response detection, in which learners may face serious consequences (e.g., academic penalties, score cancellations) based on the detection outcome.

To address this knowledge gap, we investigated the decision-making process of a custom AI-generated response detector using probing techniques. Probing has proven effective in examining the internal mechanisms of language models across a variety of downstream tasks (Li et al., 2023a; Niven and Kao, 2019; Ohmer et al., 2024; Suau et al., 2020). In this study, we fine-tuned an open-source language model on a dataset including both human-written and AI-generated responses. We then identified and manipulated lexical cues to gauge their influence on the model's classification decisions.

Our findings show that the fine-tuned model achieved high accuracy in the detection task by relying heavily on a small set of lexical cues. While this reliance demonstrates the expressive capacity of language models, it also exposes their vulnerability to exploitation and manipulation. Overall, our results substantiate the need for understanding what AI-generated response detectors learn and for evaluating the trustworthiness of their decisions in real-world applications.

2 Background

Although the language generation capabilities of modern foundation models have provided opportunities and benefits, they also pose risks that can lead to undesirable outcomes. Crothers et al. (2023) proposed a taxonomy that classifies these into four high-level categories: (1) spam and harassment, (2)

Proceedings of the Artificial Intelligence in Measurement and Education Conference (AIME-Con) – Volume 3: Coordinated Session Papers, pages 99–106

online influence campaigns, (3) malware and social engineering, and (4) AI authorship exploitation. For our study, a particularly relevant form of AI authorship exploitation is academic fraud committed by learners and examinees. They may undermine the learning and assessment purposes of writing tasks by submitting responses that are generated by AI models.

To mitigate the authorship exploitation risk, researchers have explored various detection strategies and their effectiveness. A consistent finding across studies is that humans find it difficult to reliably detect AI-generated text. Multiple investigations have shown that human judges, including domain experts, often perform at near-chance levels when attempting to distinguish AI-generated text from human-written one (e.g., Li et al., 2023b; Soni and Wade, 2023; Uchendu et al., 2021), although training (Liu et al., 2023) and auxiliary information (Gehrmann et al., 2019) may improve their detection performance. The difficulty of manual detection, combined with the scalability of AI-generated text, has led researchers to algorithmic approaches. This pursuit has quickly accumulated into a sizable body of literature, for which multiple comprehensive surveys are available (e.g., Beresneva, 2016; Crothers et al., 2023; Dhaini et al., 2023; Jawahar et al., 2020; Fraser et al., 2025; Wu et al., 2025).

Wu et al. (2025) and Fraser et al. (2025) classified algorithmic detectors into three main categories based on the type of information leveraged: watermarks, manually engineered features, and language model-based text representations. The third category uses numerical embeddings derived from foundation language models as implicit features for classification; this allows researchers to circumvent the need for watermarks or manual feature development. Detectors based on this approach, particularly those that relied on fine-tuned language models, have demonstrated strong performance, with detection accuracies often exceeding 90% across diverse text types (e.g., Chen et al., 2023; Fagni et al., 2021; Guo et al., 2023; Wang et al., 2023). However, the complexity of their architecture makes it difficult to examine their decision making processes. Although there are various linguistic differences between AI-generated and human-written texts (e.g., Seals and Shalin, 2023), it is unclear whether and how these differences are utilized by classifiers.

An effective approach for investigating the internal mechanisms of language model classifiers involves the use of probing through adversarial examples: data points that are intentionally perturbed to challenge a model's decision boundaries while preserving the original semantic content. These examples function as diagnostic tools that can help identify the specific cues that language models rely on when making classification decisions. For example, Niven and Kao (2019) demonstrated that high classification performance can be achieved through reliance on superficial word-level statistical patterns alone rather than meaningful linguistic understanding. Their work demonstrated how adversarial probing can reveal vulnerabilities in a model's generalization capabilities and shed light on its interpretability. Subsequent studies have applied adversarial probing to better understand the decision-making processes of language models fine-tuned for a range of classification tasks (e.g., Li et al., 2023a; Ohmer et al., 2024; Suau et al., 2020).

In the domain of AI-generated text detection, adversarial examples have also been used to evaluate the robustness of detection systems. These adversarial "attacks" may operate at varying levels of granularity, including character-level perturbations (e.g., Wang et al., 2024), word-level substitutions (e.g., Pu et al., 2023; Wang et al., 2024), and paraphrasing techniques that maintain semantic meaning while altering surface form (e.g., Shi et al., 2024; Krishna et al., 2023). While these studies have effectively demonstrated the vulnerability of detectors to such attacks, they often focus primarily on evasion rather than on interpretability. As a result, the internal decision-making processes of these detectors remain largely opaque.

3 Methods

3.1 Data

Our dataset included both authentic responses written by human examinees and AI-generated responses. The authentic responses were collected from an essay writing task administered as part of a standardized English language proficiency assessment. In this task, examinees were asked to express their opinion or preference on a given topic, providing supporting details. We used 5,745 authentic responses on across 20 different topics submitted by examinees representing a diverse range of nationalities and first languages. The dataset also included 6,000 responses on the same 20 topics generated by GPT-3.5 (Ouyang et al., 2022) and

GPT-4 (Achiam et al., 2023). These synthetic responses were produced as part of a separate study (Zu et al., 2025), which provides a detailed description of the generation process.

AI-generated text typically lacks typographical errors, whereas such errors are common in human-written ones, including the authentic responses in our dataset. This discrepancy could easily be exploited by detection models, potentially reducing the task to a trivial problem. To address this issue, Zu et al. (2025) randomly imputed typographical errors into each AI-generated response, and we used the generated responses that included these imputed errors.

We allocated approximately 80% of the total dataset (9,396 out of 11,745 responses) for training and the remaining 20% (2,349 responses) for testing. The train-test split involved stratified random sampling, with generation status (authentic vs. AI-generated) as the stratification variable. This ensured that both the training and test sets maintained a similar proportion of generated responses (approximately 51%).

3.2 Fine-Tuning Detector

We fine-tuned the RoBERTa-base model (Liu et al., 2019) as our primary detector of AI-generated responses. The key hyperparameters for fine-tuning included learning rate and training epochs, which were tuned through a two-dimensional grid search using five-fold cross-validation on the training set. We then used the hyperparameter values that led to the best cross-validation performance to fine-tune the RoBERTa base model using the entire training set. More details about the fine-tuning process can be found in Zu et al. (2025).

The choice of RoBERTa-base was primarily motivated by convenience. To examine the robustness of our findings with respect to this model choice, we also fine-tuned three additional models: RoBERTa-large, and two DeBERTa models (He et al., 2021) of different sizes (base and large). These alternative models were fine-tuned using the same procedure as the main detector based on RoBERTa-base.

3.3 Examining n-gram Distributions

To identify linguistic cues that our detector would learn during fine-tuning, we analyzed the n-gram distributions in authentic and AI-generated responses within the training set. For an n-gram to be considered informative, it must satisfy two conditions:

1. It should exhibit a distinct distribution between authentic and generated responses.

2. It should occur with sufficient frequency in the training data.

To quantify these conditions, we adapted the π and ξ statistics introduced by Niven and Kao (2019). Let $\mathcal{A}$ and $\mathcal{G}$ denote the sets of authentic and generated responses in the training set, respectively. Let n_{ui} represent the count of an n-gram u in response i. Using this notation, we formally introduce the two adapted metrics below.

The asymmetry metric, adapted from the π statistic in Niven and Kao (2019), captures the relative difference in frequency of u between generated and authentic responses:

$$\text{Asymmetry}_u = \frac{\sum_{i\in\mathcal{G}} n_{ui} - \sum_{j\in\mathcal{A}} n_{uj}}{\sum_{i\in\mathcal{G}} n_{ui} + \sum_{j\in\mathcal{A}} n_{uj}}.$$

This metric ranges from -1 to 1. The value of -1 indicates that the n-gram appears exclusively in authentic responses. Similarly, the value of 1 indicates exclusive presence in generated responses.

The impact metric, adapted from Niven and Kao's (2019) ξ statistic, measures the average difference in frequency per response:

$$\text{Impact}_u = \frac{\sum_{i\in\mathcal{G}} n_{ui} - \sum_{j\in\mathcal{A}} n_{uj}}{(|\mathcal{G}| + |\mathcal{A}|)/2},$$

where $|\mathcal{G}|$ and $|\mathcal{A}|$ denote the number of generated and authentic responses (in the training set), respectively. The sign of the impact metric aligns with that of the asymmetry metric, indicating the direction of distributional difference.

We analyzed the distributions of unigrams, bigrams, and trigrams in the training set using the asymmetry and impact metrics, with the goal of identifying n-grams exhibiting both high asymmetry and high impact. For the unigram analysis, 129 stop words[1] were excluded. The bigram and trigram analyses were conducted twice: once including the stop words and once excluding them. The identified n-grams were used to construct adversarial examples for probing the behavior of the fine-tuned detector.

[1]We constructed this list by adding `may` and `would` to the 127 stop words from https://gist.github.com/sebleier/554280.

4 Results

4.1 Detector Performance

The fine-tuned RoBERTa-base detector achieved an overall test set accuracy of 0.991, with a precision of 0.983 and a perfect recall of 1.0. This strong performance was robust across different model choices: each of the three alternative fine-tuned detectors achieved similarly high accuracy, precision, and recall. Table 1 presents the confusion matrices for all four fine-tuned detectors. In addition, the detector's performance remained stable under basic text manipulations. For example, converting all characters to lowercase and removing punctuation had minimal impact on accuracy, precision, or recall.

		True Label	
		Aut.	**Gen.**
RoBERTa-base	Aut.	1134	0
	Gen.	21	1194
RoBERTa-large	Aut.	1145	0
	Gen.	10	1194
DeBERTa-base	Aut.	1151	0
	Gen.	4	1194
DeBERTa-large	Aut.	1154	0
	Gen.	1	1194

Table 1: Test set confusion matrices for the main and three alternative fine-tuned detectors. Aut: Authentic; Gen.: Generated

4.2 n-gram Distributions

The results from the unigram, bigram, and trigram analyses showed notable differences in their potential utility as classification cues. The bigram and trigram distributions included only a few sequences that stood out in terms of asymmetry and impact. Moreover, most such bigrams and trigrams were composed primarily of stop words. When stop words were excluded, the same analysis yielded few prominent sequences. The unigram distributions, on the other hand, showed greater potential for distinguishing between authentic and generated responses. While most unigrams in the training set had near-zero asymmetry and impact values, a small subset had large absolute values on one or both metrics, suggesting their potential as strong indicators. This overall pattern is illustrated in Figure 1 as a bivariate scatter plot of unigram asymmetry and impact values. In addition, Table 2 lists the top 10 unigrams in terms of their absolute impact metrics.

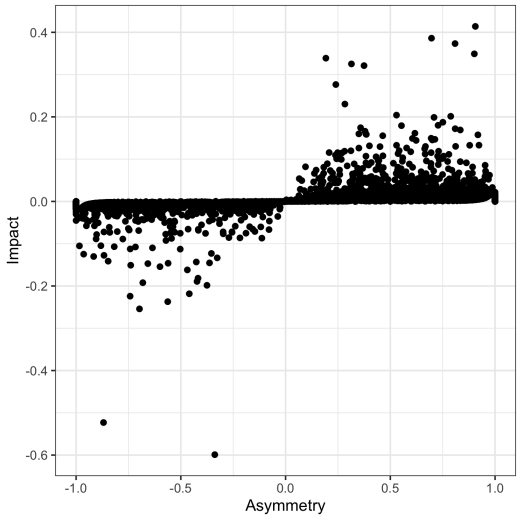

Figure 1: The asymmetry and impact metrics from the training set responses.

Table 2: Asymmetry (Asy.), impact (Imp.), and signal direction (Dir.) of the top 10 unigrams in the training set, in descending order of their absolute impact metrics.

	Asy.	Imp.	Dir.
`people`	-0.338	-0.599	Authentic
`think`	-0.869	-0.523	Authentic
`individuals`	0.906	0.414	Generated
`provide`	0.697	0.386	Generated
`overall`	0.809	0.373	Generated
`additionally`	0.901	0.349	Generated
`skills`	0.314	0.325	Generated
`learning`	0.375	0.321	Generated
`example`	-0.697	-0.254	Authentic
`good`	-0.563	-0.237	Authentic

A key distinction between unigrams associated with authentic versus generated responses was their lexical complexity or sophistication. Words that are long and typically used in formal settings tended to signal generated responses, whereas shorter, more informal ones were more indicative of authentic responses. This pattern manifested in the average length of unigrams signaling the two classes: among the 248 unigrams whose absolute asymmetry and impact values exceeded 0.05, the 72 unigrams signaling authentic responses were on average 5.0 characters long, whereas the corresponding average for the 176 unigrams signaling generated responses was 7.9. This also aligns with our prior expectations that human examinees in timed test-

ing contexts are more likely to produce draft-like responses, which may involve less frequent use of sophisticated and formal vocabulary, and that the training data for GPT 3.5 and GPT-4 are likely to consist primarily of final versions of texts rather than drafts.

4.3 Transforming Test Set Responses into Adversarial Examples

To probe the fine-tuned detector, we transformed test set responses into adversarial examples by replacing unigrams that signaled either authentic or generated responses with synonyms indicative of the opposite class. We focused on unigrams that met two criteria: (1) high absolute values on both asymmetry and impact metrics, and (2) availability of a synonym frequently used in the opposite class. For example, the unigram `people`, which strongly signaled an authentic response, was replaced with `individuals`, a word that appeared much more frequently in generated responses. To ensure meaningful substitutions, we allowed synonyms that were not unigrams, provided they occurred frequently in the opposite category. For instance, `additionally`, which appeared almost exclusively in generated responses, was replaced with `in addition`, a phrase more frequently used in authentic responses.

Applying these criteria to the training set yielded 149 unigrams to be replaced with their respective synonyms. The selected unigrams were a small subset of all unigrams in the training set, which included more than 30,000 unique unigrams. Among the 149 unigram-synonym pairs, 100 involved unigrams signaling generated responses paired with synonyms more frequent in authentic responses, while the remaining 49 involved the reverse pairing. Replacing the 149 unigrams with their synonyms resulted in, on average, 12 substitutions per response, affecting less than 10% of the average unigram count per response. This controlled transformation allowed us to evaluate the classifier's sensitivity to lexical shifts while preserving overall semantic content.

4.4 Detector Performance on Adversarial Examples

The transformation of test set responses into adversarial examples noticeably degraded the performance of the fine-tuned detector. Its overall accuracy dropped from 0.991 to 0.580. This accuracy is only slightly higher than that of a degenerate detector classifying every input into the most frequent category (whose accuracy would have been 0.508). In addition to the decline in overall accuracy, the number of responses classified as generated also dropped from 1,215 (from the original test set) to 210 on the post-transformation adversarial examples. Among those that were classified as generated, all but one response were indeed generated, resulting in a still high precision of 0.995. However, the reduced number of detected responses inevitably led to a sharp reduction in recall, which fell from being perfect (1.0) to extremely low ($209/1,194 = 0.175$), as can be seen the confusion matrix in Table 3.

		True Label	
		Aut.	**Gen.**
RoBERTa-base	Aut.	1154	985
	Gen.	1	209
RoBERTa-large	Aut.	1154	844
	Gen.	1	350
DeBERTa-base	Aut.	1154	898
	Gen.	1	296
DeBERTa-large	Aut.	1155	1097
	Gen.	0	97

Table 3: Confusion matrices for the main and three alternative fine-tuned detectors on the adversarial examples. Aut: Authentic; Gen.: Generated

The substantial decline in the frequency of responses classified as generated indicates that the detector classified much more of the adversarial examples as authentic ones than it did for the original responses. This in turn suggests that the performance change could primarily be attributed to the replacement of the 100 unigrams that were signaling generated responses. To further substantiate this conjecture, we did another transformation of the original test set responses, this time only replacing the 100 such synonym pairs while leaving the other 49 pairs unchanged. The results were quite similar as those from the full transformation involving all 149 unigrams (reported in Table 3), with the overall accuracy of 0.581 and recall of 0.175 as well as the same tendency of classifying only a small number of responses as generated. In contrast, when we did the opposite transformation of only replacing the 49 authentic-signaling synonyms, the results changed little compared to the original results (reported in Table 1): overall accuracy, precision, and recall of 0.966, 0.999, and

0.934, respectively. In sum, the performance declined primarily because the replacement of the 100 unigrams signaling generated responses tricked the detector into classifying generated responses as authentic ones.

These overall results were persistent against model choice. Table 3 also presents the confusion matrices from the three alternative detectors. All show the same pattern of substantial drop in overall accuracy, primarily attributable to the drop in the frequency of responses classified as generated and the accompanying drop in recall. This suggests that all four pre-trained language models mostly picked up unigrams signaling generated responses in the training set during fine-tuning and relied heavily on those unigrams to make their classification decisions.

5 Discussion & Conclusions

In this study, we probed an AI-generated response detector to understand how the model makes its decisions. The detector was built by fine-tuning the RoBERTa-base model (as well as three alternative language models) on a custom dataset, achieving 99.1% accuracy on a held-out test set. To identify influential lexical cues, we analyzed n-gram distributions in the training data and found 149 unigrams strongly associated with either class. By replacing these unigrams with synonyms indicative of the opposite class, we created adversarial test examples that reduced the detector's accuracy from 99.1% to 58.0%. This drop was primarily due to misclassification of AI-generated responses: altering only a small number of unigrams per response was sufficient to cause most AI-generated responses to be misclassified as authentic. The effect was consistent across all tested base models. These findings reveal the detector's strong reliance on a narrow set of lexical cues, which carries both promising and concerning implications.

On the positive side, pre-trained language models effectively identified and leveraged meaningful patterns in unigram distributions during fine-tuning, resulting in high performance on held-out data. Manually identifying these patterns would have been much more difficult and time-consuming. Moreover, such patterns can be used to build more interpretable and explainable classifiers with minimal loss in performance, assuming the patterns remain stable in future data.

However, the ease with which the detector's accuracy was reduced to near-chance levels raises concerns about its generalizability and robustness. If the small set of unigrams signaling AI-generated responses becomes widely known, malicious actors could evade detection by substituting a few words, as demonstrated in our adversarial examples. Therefore, a promising direction for future research is to devise ways to encourage detectors to learn more robust patterns. The identification of this major concern and promising future research step underscore the value of probing fine-tuned detectors in understanding what they learn, evaluating the trustworthiness of their decisions in real-world applications, and guiding improvements where necessary.

We acknowledge that this study was limited in its scope. All detectors were trained on responses from a single task type covering a relatively narrow set of 20 topics. Large-scale writing tests, on the other hand, may include multiple task types and a broader range of topics to ensure topical diversity and coverage. A training dataset drawn from such varied sources may exhibit different characteristics than those observed in our study, and the robustness of detectors trained on more diverse data cannot be reliably inferred from our findings. Furthermore, even within similar training contexts, the rapid evolution of generative AI raises uncertainty about whether the same lexical cues will remain effective indicators of AI-generated content. Therefore, our findings should be interpreted primarily as evidence of what fine-tuned detectors can learn, and how easily they can be compromised, rather than as prescriptive guidance for detection or evasion strategies.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Daria Beresneva. 2016. Computer-generated text detection using machine learning: A systematic review. In *International Conference on Applications of Natural Language to Information Systems*, pages 421–426. Springer.

Yutian Chen, Hao Kang, Vivian Zhai, Liangze Li, Rita Singh, and Bhiksha Raj. 2023. GPT-sentinel: Distinguishing human and ChatGPT generated content. *arXiv preprint arXiv:2305.07969*.

Evan N Crothers, Nathalie Japkowicz, and Herna L Viktor. 2023. Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, 11:70977–71002.

Mahdi Dhaini, Wessel Poelman, and Ege Erdogan. 2023. Detecting ChatGPT: A survey of the state of detecting ChatGPT-generated text. *arXiv preprint arXiv:2309.07689*.

Tiziano Fagni, Fabrizio Falchi, Margherita Gambini, Antonio Martella, and Maurizio Tesconi. 2021. TweepFake: About detecting deepfake tweets. *Plos one*, 16(5):e0251415.

Kathleen C Fraser, Hillary Dawkins, and Svetlana Kiritchenko. 2025. Detecting AI-generated text: Factors influencing detectability with current methods. *Journal of Artificial Intelligence Research*, 82:2233–2278.

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

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. *arXiv preprint arXiv:2301.07597*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTaV3: Improving DeBERTa using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. 2020. Automatic detection of machine generated text: A critical survey. *arXiv preprint arXiv:2011.01314*.

Yang Jiang, Jiangang Hao, Michael Fauss, and Chen Li. 2024. Detecting ChatGPT-generated essays in a large-scale writing assessment: Is there a bias against non-native english speakers? *Computers & Education*, 217:105070.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36:27469–27500.

Jiawei Li, Yizhe Yang, Yu Bai, Xiaofeng Zhou, Yinghao Li, Huashan Sun, Yuhang Liu, Xingpeng Si, Yuhao Ye, Yixiao Wu, and 1 others. 2024. Fundamental capabilities of large language models and their applications in domain scenarios: A survey. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11116–11141.

Jiaxuan Li, Lang Yu, and Allyson Ettinger. 2023a. Counterfactual reasoning: Testing language models' understanding of hypothetical scenarios. *arXiv preprint arXiv:2305.16572*.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023b. MAGE: Machine-generated text detection in the wild. *arXiv preprint arXiv:2305.13242*.

Yikang Liu, Ziyin Zhang, Wanyang Zhang, Shisen Yue, Xiaojing Zhao, Xinyuan Cheng, Yiwen Zhang, and Hai Hu. 2023. ArguGPT: evaluating, understanding and identifying argumentative essays generated by GPT models. *arXiv preprint arXiv:2304.07666*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*.

Xenia Ohmer, Elia Bruni, and Dieuwke Hupke. 2024. From form(s) to meaning: Probing the semantic depths of language models using multisense consistency. *Computational Linguistics*, 50(4):1507–1556.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Jiameng Pu, Zain Sarwar, Sifat Muhammad Abdullah, Abdullah Rehman, Yoonjin Kim, Parantapa Bhattacharya, Mobin Javed, and Bimal Viswanath. 2023. Deepfake text detection: Limitations and opportunities. In *2023 IEEE symposium on security and privacy (SP)*, pages 1613–1630. IEEE.

Spenser M Seals and Valerie L Shalin. 2023. Long-form analogies generated by ChatGPT lack human-like psycholinguistic properties. *arXiv preprint arXiv:2306.04537*.

Zhouxing Shi, Yihan Wang, Fan Yin, Xiangning Chen, Kai-Wei Chang, and Cho-Jui Hsieh. 2024. Red teaming language model detectors with language models. *Transactions of the Association for Computational Linguistics*, 12:174–189.

Mayank Soni and Vincent Wade. 2023. Comparing abstractive summaries generated by ChatGPT to real summaries through blinded reviewers and text classification algorithms. *arXiv preprint arXiv:2303.17650*.

Xavier Suau, Luca Zappella, and Nicholas Apostoloff. 2020. Finding experts in transformer models. *arXiv preprint arXiv:2005.07647*.

Adaku Uchendu, Zeyu Ma, Thai Le, Rui Zhang, and Dongwon Lee. 2021. Turingbench: A benchmark environment for turing test in the age of neural text generation. *arXiv preprint arXiv:2109.13296*.

Yichen Wang, Shangbin Feng, Abe Bohan Hou, Xiao Pu, Chao Shen, Xiaoming Liu, Yulia Tsvetkov, and Tianxing He. 2024. Stumbling blocks: Stress testing the robustness of machine-generated text detectors under attacks. *arXiv preprint arXiv:2402.11638*.

Zecong Wang, Jiaxi Cheng, Chen Cui, and Chenhao Yu. 2023. Implementing BERT and fine-tuned RoBERTa to detect AI generated news by ChatGPT. *arXiv preprint arXiv:2306.07401*.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. 2025. A survey on LLM-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, 51(1):275–338.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, and 1 others. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).

Jiyun Zu, Michael Fauss, and Chen Li. 2025. Effects of generation model on detecting AI-generated essays in a writing test. In *Artificial Intelligence in Measurement and Education Conference*. NCME.