

‘Quis custodiet ipsos custodes?’ *Who will watch the watchmen?* On Detecting AI-generated peer-reviews

Anonymous EMNLP submission

Abstract

The integrity of the peer-review process is vital for maintaining scientific rigor and trust within the academic community. With the steady increase in the usage of large language models (LLMs) like ChatGPT in academic writing, there is a growing concern that AI-generated texts could compromise the scientific publishing including peer-reviews. Previous works have focused on generic AI-generated text detection or have presented an approach for estimating the fraction of peer-reviews that can be AI-generated. Our focus here is to solve a real-world problem by assisting the editor or chair in determining whether a review is written by ChatGPT or not. To address this, we introduce the Term Frequency (TF) model, which posits that AI often repeats tokens, and the Review Regeneration (RR) model which is based on the idea that ChatGPT generates similar outputs upon re-prompting. We stress test these detectors against token attack and paraphrasing. Finally we propose an effective defensive strategy to reduce the effect of paraphrasing on our models. Our findings suggest both our proposed methods perform better than other AI text detectors. Our RR model is more robust, although our TF model performs better than the RR model without any attacks. We make our code, dataset, model public¹.

1 Introduction

Large language models (LLMs), such as ChatGPT, PaLM (Chowdhery et al., 2023) and GPT-4 (Achiam et al., 2023), have significantly impacted both the industrial and academic sectors. The surge in Artificial Intelligence (AI)-generated content has permeated various domains, from journalism (Gutiérrez-Caneda et al., 2023; Shi and Sun, 2024) to academia (Bin-Nashwan et al., 2023; Shi et al., 2023). However, their misuse also introduces concerns—especially regarding fake news (Zhang and

Gao, 2023; Silva and Vaz, 2024), fake hotel reviews (Ignat et al., 2024), fake restaurant review (Gambetti and Han, 2024). The exceptional human like fluency and coherence of the generated content of these models pose a significant challenge, even for experts, in distinguishing if the text is written by human or LLMs (Shahid et al., 2022).

What if peer-reviews themselves are AI-generated?

A study (Liang et al., 2024a) conducted experiments on a few papers of AI conferences and found that between 6.5% and 16.9% of text submitted as peer-reviews to these conferences could have been substantially modified by LLMs. They estimated that the usage of ChatGPT in reviews increases significantly within three days of review deadlines. Reviewers who do not respond to ICLR/NeurIPS author rebuttals exhibit a higher estimated usage of ChatGPT. Additionally, an increase in ChatGPT usage is associated with low self-reported confidence in reviews. Once Springer retracted 107 cancer papers after they discovered that their peer-review process had been compromised by fake peer-reviewers (Chris Graf, 2022).

Previous works have focused on studying the effect of ChatGPT on AI conference peer-reviews. However, in this paper, our focus is to determine whether a review is written by ChatGPT or not. In this paper we do not assert that AI-generated peer-reviews inherently detract from the quality or integrity of the peer-review system. There can be debates whether AI-generated reviews can help peer-review system or not. But, in this paper we are not asserting that AI-generated peer-review is completely not useful. However we believe if the review is AI-generated chair/meta reviewer should be well aware. It is breach of trust if meta reviewer believes that the review is a human-written nevertheless it is not. Despite the potential benefits

¹<https://anonymous.4open.science/r/AI-Review-Detection-5544>

AI-generated reviews may offer, it is crucial for editors to exercise discernment in their reliance on these reviews. This caution is warranted due to the intrinsic limitations of current language models, which can produce inaccurate, misleading (Pan et al., 2023), or entirely fabricated information—a phenomenon often referred to as hallucination (Ji et al., 2023; Rawte et al., 2023).

In this paper, we propose two simple yet effective methods to detect AI-generated peer-review based on token frequency (TF method) and regeneration based approach (RR method). We also propose a token modification attack method based and studied its effect on various detectors. Paraphrasing attack is very common way to evade the text detection. So, we also study the effect of paraphrasing on various text detectors. Finally we propose a technique to defend our regeneration based technique to the paraphrasing attack. We found that both TF model and the RR model perform better than other AI text detectors for this task. We also found that while TF model performs better than the RR model under normal conditions while RR model is more robust and is able to withstand adjective attacks and paraphrasing attacks (after the defense is applied).

OUR CONTRIBUTIONS:

- We introduce a novel task to address the real-world problem of detecting AI-generated peer-reviews. We create a novel dataset of 1,480 papers from the ICLR and NeurIPS conferences for this task.
- We propose two techniques, namely the token frequency-based approach (TF) and the regeneration-based approach (RR), which perform better than existing AI text detectors.
- We stress-test the detectors against token attacks and paraphrasing, and propose an effective defensive strategy to reduce evasion during paraphrasing attacks.

2 Related Work

2.1 Zero-Shot Text Detection

Zero-shot text detection does not require training on specific data and directly identifies AI-generated text using the model that produced it (Mitchell

et al., 2023). (Solaiman et al., 2019) use average log probability of a text under the generative model for detection, whereas DetectGPT (Mitchell et al., 2023) uses property of AI text to occupy negative curvature regions of model’s log probability function. Fast-DetectGPT (Bao et al., 2023a) increases its efficiency by putting conditional probability curvature over raw probability. (Tulchinskii et al., 2023) showed that average intrinsic dimensionality of AI-generated texts is lower than the human. The paper (Gehrmann et al., 2019) estimate probability of individual tokens and detect AI-generated text by applying threshold on probability.

2.2 Training based Text Detection

Some researchers have fine-tuned language models to recognize LLM-generated text. (Guo et al., 2023) trained OpenAI text classifier on a collection on millions of text. GPT-Sentinel (Chen et al., 2023) train RoBERTa (Liu et al., 2019) and T5 (Raffel et al., 2020) classifiers on OpenGPT-Text. LLM-Pat (Yu et al., 2023) trained a neural network on the similarity between candidate texts and reconstructed sibling text generated by an intermediary LLM (parent). But due to excessive reliance of this model on training data many models show vulnerability to adversarial attacks (Wolff, 2020).

2.3 LLM Watermarking

The concept of watermarking AI-generated text, initially introduced by (Wiggers, 2022), involves embedding an undetectable signal to attribute authorship to a particular text with a high level of confidence which is similar to encryption and decryption. In simple words a watermark is a hidden pattern in text that is imperceptible to humans. It involves adding some kind of pattern which can be recognized by algorithms directly into the text and some techniques also involve integrating an machine learning model in the watermarking algorithm itself (Abdelnabi and Fritz, 2021; Munyer and Zhong, 2023; Yoo et al., 2023; Qiang et al., 2023).

Watermarked text can be generated using a standard language model without re-training (Kirchenbauer et al., 2023). It planted watermarks with large enough entropy, resulting change in distribution of generated texts. (Zhao et al., 2023) proposed a method of injecting secret sinusoidal signals into decoding steps for each target token. However, (Singh and Zou, 2023) addresses the issue that watermarking can compromise text generation

quality, coherence and depth of LLM responses. (Chakraborty et al., 2023a) suggests that watermarked texts can be circumvented and paraphrasing does not significantly disrupt watermark signals thus text watermarking is fragile and lacks reliability for real-life applications.

2.4 Statistical Estimation Approach

There have been inquiries into the theoretical feasibility of achieving precise detection on an individual level (Weber-Wulff et al., 2023; Sadasivan et al., 2023a; Chakraborty et al., 2023b). (Liang et al., 2024a) presented an approach for estimating the fraction of text in a large corpus using a maximum likelihood estimation of probability distribution without performing inference on an individual level thus making it computationally efficient. They conducted experiments on papers from a few AI conferences to determine the fraction of peer-reviews that could have been substantially modified by LLMs.

2.5 AI-generated Research Paper Detection

The DagPap22 Shared Task (Kashnitsky et al., 2022) aimed to detect automatically generated scientific papers. The winning team (Rosati, 2022) utilized a DeBERTa v3 model that was fine-tuned on their dataset (almost all teams managed to surpass the baseline models, Tf-IDF and logistic regression). It was also concluded that machine-generated text detectors should not be used in production because they perform poorly with distribution shifts, and their effectiveness on realistic full-text scientific manuscripts remains untested.

3 Dataset

We collected a total of 1,480 papers from Open-Review Platform². The first version of ChatGPT was released by OpenAI on November 30, 2022. Therefore, we choose papers from 2022, ensuring there was almost no chance that any of the collected reviews were already generated by ChatGPT.

Figure 1 shows the overall statistics of AI-generated reviews and golden reviews for both ICLR and NeurIPS reviews. We discuss the creation of the dataset in more details in the Appendix Section A. We split the dataset into 70%, 15%, and 15% for training validation and test set respectively.

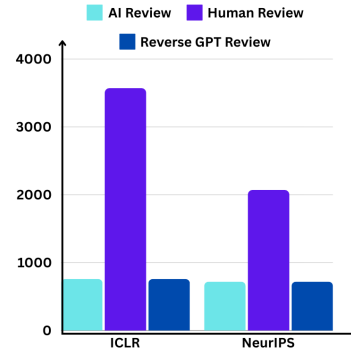


Figure 1: Dataset Statistics. Here, x axis: Different Venue ; y axis: Number of reviews

4 Methodology

In this section, we present our two approaches to detect AI written peer-reviews, based on token frequency (Section 4.1) and review regeneration (Section 4.2). Then, we propose a possible attack (Token Manipulation Attack) on the AI text detectors to see how various models react on it in Section 4.3. Additionally, since paraphrasing is a common method used to circumvent AI text detection, we introduce a countermeasure as described in Section 4.4, designed to protect our proposed Review Regeneration method against such attacks.

4.1 Token Frequency based Approach

Inspired by (Liang et al., 2024b), we propose a method that utilizes the frequency of tokens within review texts. This approach is premised on the hypothesis that different types of reviews (human-generated vs. AI-generated) exhibit distinct patterns in the usage of certain parts of speech, such as adjectives, nouns, and adverbs.

Let H denote the human corpus, consisting of all human-generated reviews, and A represent the AI corpus, comprising of all AI-generated reviews. Define x as an individual review, and t as a token. This token t can be adjective or noun or adverb. To identify if token is adjective or noun or adverb, we have used the PoS-tagger of Natural Language Tool Kit (NLTK) module³.

We define $p^A(t)$ and $p^H(t)$ as the probabilities of token t appearing in the AI and human corpora, respectively. These are estimated as follows:

$$p^A(t) = \frac{\text{Count of reviews with } t \text{ in } A}{\text{Total \# of reviews in } A}$$

²<https://openreview.net/>

³<https://www.nltk.org/book/ch05.html>

$$p^H(t) = \frac{\text{Count of reviews with } t \text{ in } H}{\text{Total \# of reviews in } H}$$

Now, for each review x , we calculate $P^A(x)$ and $P^H(x)$, which represent the probability of x belonging to the AI corpus and the human corpus, respectively. These probabilities can be calculated by summing up the probabilities of all tokens that are coming in review x :

$$P^A(x) = p^A(t_1) + p^A(t_2) + \dots = \sum_{i=1}^{i=n_a} p^A(i)$$

$$P^H(x) = p^H(t_1) + p^H(t_2) + \dots = \sum_{i=1}^{i=n_h} p^H(i)$$

Here, t_1, t_2, \dots refer to the tokens occurring in review x . Also, n_a and n_h refer to the number of AI and Human corpus reviews, respectively.

If review x contains tokens with higher probabilities in the AI corpus, then $P^A(x)$ will be greater, increasing the likelihood that x is AI-generated. Conversely, if x contains tokens with higher probabilities in the human corpus, then $P^H(x)$ will be greater, suggesting that the review is more likely to be human-written.

To classify each review x_i , we calculate $p^A(i)$ and $p^H(i)$ for each review in our dataset. These serve as input features for training a neural network. The neural network is trained to distinguish between AI-generated and human-generated reviews based on these input features. By learning from the patterns and distributions of these probabilities, the neural network can accurately detect AI-generated reviews.

4.2 Regeneration based Approach

Figure 2 shows the overall architectural diagram of our proposed regeneration-based approach. The input to the framework is the paper and its review which we aim to determine whether they are written by AI or human.

The idea behind this approach is that if a similar prompt is given repeatedly to a large language model (LLM), the LLM is likely to generate reviews or responses that exhibit a consistent style, tone, and content, as outlined in the provided context. This consistency occurs because a large language model generally applies the patterns it has

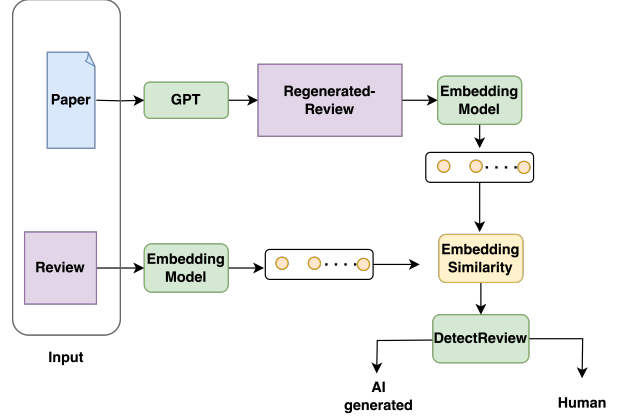


Figure 2: Architectural diagram of Regeneration based Approach

learned during training to the new content it generates based on the given prompt. The study in (Hackl et al., 2023) found that GPT-4 demonstrated high inter-rater reliability, with ICC scores ranging from 0.94 to 0.99, in rating responses across multiple iterations and time periods (both short-term and long-term). This indicates consistent performance when given the same prompt. Furthermore, the results showed that different types of feedbacks (content or style) did not affect the consistency of GPT-4's ratings, further supporting the model's ability to maintain a consistent approach based on the prompt.

4.2.1 Review Regeneration and Embedding Creation

We employ GPT to regenerate a review R^{reg} using the prompt P^{reg} . We create two distinct embeddings E_R for R^{reg} and E_F for R (review which we have to determine if the review is AI-generated or not). The idea is that if the review R is generated by an AI, we hypothesize that its embedding E_F will exhibit a closer similarity to E_R , the embedding of a known AI-generated review R^{reg} .

Then, we quantify the similarity between the embeddings using the cosine similarity metric, as outlined below:

$$\text{CosineSimilarity}(E_R, E_F) = \frac{E_R \cdot E_F}{\|E_R\| \|E_F\|}$$

Here, \cdot represents the dot product, and $\|R\|$ and $\|F\|$ represent the Euclidean norms of the embeddings. This formula calculates the cosine of the angle between the two embeddings E_R and E_F , providing a measure of similarity where values closer to 1 indicate higher similarity and thus a greater likelihood that both reviews are AI-generated.

4.2.2 Training

Next, we utilize the computed similarity score as input to train a neural network aimed at detecting AI-generated reviews. The training process involves optimizing the network’s parameters via backpropagation. This optimization is directed by the cross-entropy loss function.

4.3 Token Attack



Figure 3: AI text undetectability attack

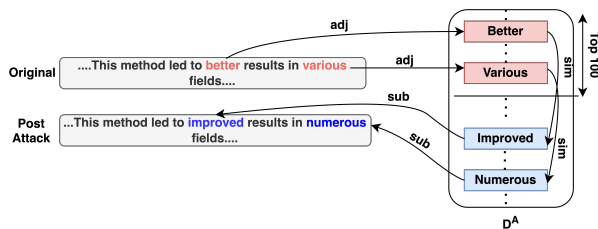


Figure 4: An example of adjective token attack. Here, sub: substitution, adj: Adjective, sim: similar token , D^A : AI word dictionary (sorted high-top to bottom-low).

We propose an attack method to reduce the probability of reviews being classified as AI-generated described in Algorithm-1 where we target the most frequent tokens in AI-generated reviews and replace them with their synonyms, which are less frequent in the AI-generated content.

Here, we focus exclusively on adjectives, referring to this approach as the "adjective attack." We chose adjectives because substituting nouns and adverbs with their synonyms often leads to nonsensical statements or drastically alters the meaning of the review. We discuss this in detail in Appendix C.

In the adjective attack, we substitute the top 100 highest probability adjective tokens (e.g., "novel," "comprehensive") with their synonyms.

To obtain synonyms for the selected tokens, we utilize the NLTK WordNet database⁴. To preserve the original meaning of tokens as much as possible, we ensure that any synonym used to replace a token

⁴<https://www.nltk.org/api/nltk.corpus.reader.wordnet>

is also present in the AI corpus. If a suitable synonym is not found in the corpus, we do not replace the token.

Algorithm 1 Token Attack

- 1: Identify top 100 high-probability tokens: w_1, w_2, \dots, w_{100} .
- 2: Retrieve synonyms for each token: $sw_1, sw_2, \dots, sw_{100}$.
- 3: Perform PoS tagging for each review
- 4: Replace each tagged token with its synonym if it matches with one of the top 100 tokens.

In order to determine which tokens from the review should be replaced with their synonyms, we performed PoS tagging on the review. For example, if we are conducting an adjective attack, we replace only the adjective tokens in the review with their synonyms.

We also illustrate this with an example of an adjective attack, as shown in Figure 4. In this example, the adjective tokens ‘better’ and ‘various’ from a review are among the top 100 AI token list. We replace them with their synonyms ‘improved’ and ‘numerous,’ respectively.

4.4 Paraphrasing Defence

Paraphrasing tools are effective in evading detection (Sadasivan et al., 2023b; Krishna et al., 2024). To increase the robustness of Regeneration based text detector to paraphrase attacks, we introduce a simple defense that employs a targeted synonym replacement strategy. The core idea behind this approach is that when an AI-generated review is processed by a paraphraser, one of the major modifications it makes is substituting the original words with the similar ones. We propose a technique to revert the paraphrased reviews back to a state that closely resembles their original AI-generated form by utilizing the regenerated review (as they would be close to the original AI-generated review).

As discussed in Algorithm-2, first, we identify all the tokens within a review and their corresponding regenerated reviews using the PoS tagging⁵. Here token can be any word in a review which are adjective, noun, or adverb. For each token in a review, we obtain a list of synonyms from the NLTK WordNet database. Then, for each synonym in that list, we check whether it is present in the

⁵We used tagger of the NLTK model. As we also discussed in Section 4.3

Algorithm 2 Paraphrasing Defence

- 1: Identify tokens in the review and regenerated reviews
 - 2: **for** each token in the review **do**
 - 3: Get synonyms of the token
 - 4: **for** each synonym in synonyms **do**
 - 5: **if** synonym is in regenerated reviews
 - 6: **then**
 - 7: Replace the token with synonym
 - 8: Break
 - 9: **else**
 - 10: Do not replace the token
-

378 corresponding regenerated review or not. If it is,
379 we replace the original token with its synonym.

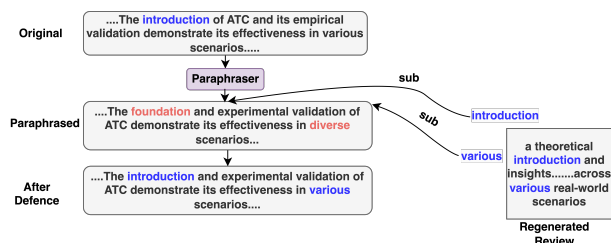


Figure 5: An example of paraphrasing defence; Here,sub: substitution.

380 We also illustrate this by an example in Figure
381 5. The paraphraser has changed the structure of the
382 sentence and also replaced some of the words like
383 ‘introduction’ with ‘foundation’, ‘empirical’ with
384 ‘experimental,’ and ‘various’ with ‘diverse’. Now,
385 after applying the defence algorithm the words
386 ‘foundation’ and ‘diverse’ gets reverted back to
387 ‘introduction’ and ‘various’, thus making it more
388 identical to its original sentence. We called a re-
389 view converted by using this algorithm as ‘modified
390 review’.

391 **Training:** In a real-world scenario, whether a re-
392 view has been paraphrased or not will be unknown,
393 and detecting this becomes a task in itself. However,
394 the aim of this paper is to propose a model that is ro-
395 bust to any kind of text, whether paraphrased or not.
396 Therefore, we retrained both models. The modified
397 training set consists of the original training set after
398 being processed by the defense algorithm. Simi-
399 larly, the modified paraphrased set consists of the
400 paraphrased reviews from the original training set,
401 which have been modified using the defense algo-
402 rithm. For testing or validation, it will be unclear
403 whether a review is paraphrased by AI or simply AI-
404 written. Therefore, we combined both the testing

405 set and the paraphrased set. Both will be modi-
406 fied by the defense algorithm before undergoing
407 validation or testing⁶.

5 Experiments 408

5.1 Experimental Settings 409

410 We implemented our system using PyTorch (Paszke
411 et al., 2019). The dataset was randomly split into
412 three parts: 80% for training, 10% for validation,
413 and 10% for testing.

414 For the TF model and RR model, we conducted
415 experiments with different network configurations
416 during the validation phase. Through these experi-
417 ments, we determined that a batch size of 32 and a
418 dropout rate of 0.1 for every layer yielded optimal
419 performance. The activation function ReLU was
420 used in our model. We trained the model for 20
421 epochs, employing a learning rate of 1e-3 for TF
422 model and 0.01 for RR model and cross-entropy
423 as the loss function. To prevent overfitting, we
424 used the Adam optimizer with a weight decay of
425 1e-3. All models were trained on an NVIDIA A100
426 40GB GPU. We used the text-embedding-ada-002⁷
427 pretrained model from OpenAI for creating em-
428 beddings of the reviewer’s review and the regen-
429 erated review. All models were trained on a single
430 NVIDIA A100 40GB GPU.

5.2 Baselines for Comparison 431

432 **RADAR (Hu et al., 2023)** (Robust AI text De-
433 tection via Adversarial Learning) draws inspira-
434 tion from adversarial machine learning techniques.
435 **LLMDet (Wu et al., 2023)** (A Third Party Large
436 Language Models Generated Text Detection Tool)
437 is a text detection tool that can identify the source
438 from which the text was generated, such as Human,
439 LLaMA, OPT, or others. **DEEP-FAKE (Li et al.,
440 2023)** Text Detection considered 10 datasets cover-
441 ing a wide range of writing tasks (e.g., story gen-
442 eration, news writing and scientific writing) from
443 diverse sources (e.g., Reddit posts and BBC news),
444 and applied 27 LLMs (e.g., OpenAI, LLaMA,
445 and EleutherAI) for construction of deepfake texts.
446 **Fast-Detect GPT (Bao et al., 2023b)** uses a condi-
447 tional probability function and it invokes the sam-
448 pling GPT once to generate all samples and calls
449 the scoring GPT once to evaluate all the samples.

⁶As a result the size of the training set will increase three-
fold, and the testing and validation sets will double

⁷[https://platform.openai.com/docs/guides/
embeddings](https://platform.openai.com/docs/guides/embeddings)

Model	Precision		Recall		F1 - Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
RADAR	66.48	66.97	75.13	81.11	70.54	73.37	66.12	69.01
LLMDET	54.69	53.24	98.42	98.06	70.30	69.01	55.11	53.65
DEEP-FAKE	93.98	93.64	92.50	91.94	93.24	92.78	89.45	88.89
FAST DETECT	95.96	94.87	81.32	66.81	88.03	78.40	88.07	80.63
Our TF Model	99.99	99.99	99.80	99.30	99.89	99.65	99.92	99.82
Our RR Model	99.32	93.75	94.38	93.10	96.79	93.43	98.67	97.24

Table 1: Comparison results of the proposed Review Regeneration technique and Token Frequency technique. Here, the AI-generated reviews and regenerated reviews are generated by GPT-4; RR: Review Regeneration, TF: Token Frequency

Model	Precision		Recall		F1-Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
RADAR	14.58	15.13	40.38	51.11	21.43	23.35	47.97	48.99
LLMDET	50.17	52.53	95.39	93.75	65.76	67.33	50.33	52.88
DEEP-FAKE	68.42	47.37	17.11	93.06	27.37	11.04	54.61	49.65
FAST DETECT	71.43	20.00	03.47	00.69	06.62	01.34	51.04	48.96
Our TF Model	99.99	99.99	11.18	05.56	20.12	10.53	81.45	79.35
Our RR Model	81.67	80.87	64.47	64.58	72.06	71.81	89.78	89.23

Table 2: Comparison results after Token Attack (Adjective)

We discuss them in details in Section D.

5.3 Results and Analysis

Table 1 shows the comparison results of the models when reviews are generated by GPT-4. It is evident from the results that our proposed TF and RR models outperform the other text detectors. Our Token Frequency (TF) model surpasses the closest comparable model DEEP-FAKE, with margins of 6.75 and 6.87 F1 points for ICLR and NeurIPS papers, respectively. It surpasses FAST DETECT by margins of 11.96 and 6.87 F1 points for ICLR and NeurIPS, respectively, RADAR by margins of 29.45 and 26.28 F1 points for ICLR and NeurIPS, respectively, and LLMDET by margins of 29.69 and 30.64 F1 points for ICLR and NeurIPS, respectively. Our Review Regeneration (RR) model outperforms DEEP-FAKE by 3.55 and 0.65 F1 points, RADAR by margins of 26.25 and 20.06 F1 points, LLMDET by 26.49 and 24.42 F1 points, DEEP-FAKE by 3.55 and 0.65 F1 points, and FAST DETECT by 8.76 and 15.03 F1 points for ICLR and NeurIPS respectively. Additionally, it is apparent from the results that our TF model outperforms our RR model by 2.51 and 6.52 F1 points for the ICLR and NeurIPS datasets respectively.

In the results reported above for the TF model, we considered tokens as adjectives, as this configuration yielded the best results. We also present the outcomes of the TF model when trained with tokens considered as adverbs or nouns in the Appendix Table 7. Furthermore, we observe a similar distribution of results on reviews generated by GPT-3.5. We report the result in Appendix Table 5.

5.3.1 Effect of attacking AI-generated text detectors using Adjective Attack

We report the results after performing adjective attack as described in Section 4.3 in Table 2. It is evident from the table that the result of each model dropped after the attack. In particular, the F1 score of RADAR dropped by 69.62% and 68.18% for ICLR and NeurIPS, respectively, LLMDET dropped by 6.46% and 2.43% for ICLR and NeurIPS, respectively, DEEP-FAKE dropped by 70.65% and 88.10% for ICLR and NeurIPS, respectively, and FAST DETECT dropped by 92.48% and 98.29% for ICLR and NeurIPS, respectively. Additionally, the F1 score of our TF model dropped by 79.88% and 89.43% for ICLR and NeurIPS, respectively, whereas for our RR model, it dropped by 25.56% and 23.14% for ICLR and NeurIPS, respectively.

The results reveal that this attack has significantly compromised the performance of our TF model, underscoring its vulnerability and limited resilience to such threats. The substantial decline in the F1-score can be attributed primarily to the model’s reliance on token frequency patterns in AI-generated reviews. These patterns are effectively disrupted by synonym replacements, leading to the observed performance degradation. After the adjective attack, we observed that our RR model outperforms other AI text detectors, including our proposed TF model, achieving the highest F1 score of 71.81.

Model	Precision		Recall		F1-Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
RADAR	88.82	95.83	51.92	53.08	65.53	68.32	53.29	55.56
LLMDet	98.68	99.31	49.83	50.00	66.23	66.51	49.67	50.00
DEEP-FAKE	83.55	78.47	70.17	60.75	76.28	68.48	74.01	63.89
FAST DETECT	59.35	57.64	48.03	60.58	53.09	59.07	71.59	73.00
Our TF Model	97.67	97.96	27.63	33.33	43.08	49.74	63.49	66.32
Our RR Model	51.92	52.75	35.53	32.43	42.19	40.17	51.32	50.86
Our TF Model (D)	76.92	64.29	74.19	84.38	75.53	72.97	95.40	93.73
Our RR Model (D)	90.87	93.98	78.62	81.25	84.30	87.15	91.51	92.86

Table 3: Comparison results after paraphrasing. Here D denotes the result after applying our proposed paraphrasing defence

5.3.2 Effect of attacking AI-generated text detectors using Paraphrasing Attack

Next, we report the result after performing paraphrasing (See Appendix E for more details) on the AI-generated reviews. It is evident from the Table 3 that the result of each model dropped after the attack. In particular, the F1 score of RADAR dropped by 7.10% and 6.89% for ICLR and NeurIPS, respectively, LLMDet dropped by 5.79% and 3.62% for ICLR and NeurIPS, respectively, DEEP-FAKE dropped by 18.19% and 26.19% for ICLR and NeurIPS respectively, and FAST DETECT dropped by 39.69% and 24.66% for ICLR and NeurIPS respectively. Additionally, F1 score of our TF model dropped by 56.92% and 50.08% for ICLR and NeurIPS respectively and RR model dropped by 56.41% and 57.00% for ICLR and NeurIPS respectively.

This effect on the TF model is not surprising, as it is based on AI token frequency and paraphrasing typically involves replacing words with their synonyms. For our RR model, we noted that paraphrasing caused both human-written and AI-written reviews to diverge further from the regenerated reviews. This increased dissimilarity could stem from various factors, including alterations in text structure, voice, tone, and vocabulary. If only human reviews had been paraphrased, we might have observed an improvement in performance due to a greater distinction between human-written and regenerated reviews. In our test set, which includes both AI-generated and human reviews, the similarity of AI-generated text decreased following paraphrasing, leading to a decline in overall performance.

5.3.3 Results after Paraphrasing Defence

Next, we report the result after performing paraphrasing Defence (See Section 4.4 for more details) on both our proposed models on Table 3. We observed improvements in both our TF and RR mod-

els. We also applied the defense to other AI text detection algorithms and observed no significant improvement or decrease in their results. These results are reported in Table 8. The performance of the TF model improved by 75.32% for ICLR papers and 46.70% for NeurIPS. Similarly, the performance of the RR model improved by 99.81% for ICLR and 111.69% for NeurIPS.

These results indicates that our proposed RR model is more robust against different types of attacks and performs better than any other existing text detection algorithms.

5.4 Human evaluation

We also conducted human analyses to understand when and why our models fails. We discuss this extensive error analysis in the Appendix B.

6 Conclusion and Future Work

In this work, we propose two methods to determine whether a review is written by a human or generated by AI. We found that our proposed TF model and the RR model outperform other AI text detectors under normal conditions. We stress test these detectors against token attack and paraphrasing. Furthermore, our proposed RR model is more robust and outperforms other methods. We then propose an effective defensive strategy to reduce the effect of paraphrasing on our models. Our findings suggest both our proposed methods perform better than other AI text detectors. Also, while our proposed TF model performs better than the RR model without any attacks, our RR model is more robust against token attacks and paraphrasing attacks.

We hope that these findings will pave the way for more sophisticated and reliable AI detectors to prevent such misuse. For further work, we aim to focus on cases where the reviewer writes parts of the review using AI.

592 Limitations

593 Our study primarily utilized GPT-4 and GPT-3.5
594 for generating AI texts, as GPT has been one of
595 the most widely used LLMs for long-context con-
596 tent generation. We recommend that future practi-
597 tioners choose the LLM that best aligns with the
598 language model likely used to generate their tar-
599 get corpus, to accurately reflect usage patterns at
600 the time of its creation. Our methods are specifi-
601 cally designed for reviews completely written by
602 AI. It is possible, however, that a reviewer may
603 outline several bullet points related to a paper and
604 use ChatGPT to expand these into full paragraphs.
605 We suggest exploring this aspect in future research.

606 Ethics Statement

607 We have utilized the open source dataset for our
608 work. Our Institutional Review Board (IRB) evalu-
609 ated and approved this study. We emphasize here
610 that we do not wish to pass a judgement or claim
611 that the use of AI tools for review papers is neces-
612 sarily bad or good. We do not provide definitive
613 proof that reviewers are employing ChatGPT to
614 draft reviews. It is important to emphasise that the
615 primary purpose of this system is to assist editors
616 by highlighting reviews that can be potentially AI-
617 generated ; this system is only for editors' internal
618 usage, not for authors or reviewers.

619 Our RR model requires regenerated review to
620 be generated from paper using LLM. Also, open-
621 sourced LLMs running locally will not have any
622 concerns. OpenAI implemented a Zero Data Re-
623 tention policy to ensure the security and privacy of
624 data. Additionally, users can control the duration of
625 data retention through ChatGPT Enterprise⁸. Also,
626 nowadays, many papers are submitted to arXiv and
627 are publicly available⁹. However, editors and chairs
628 should use this tool with caution, considering the
629 potential risks to privacy and anonymity.

630 The system does not detect all AI-generated re-
631 views. Given its nature as a general AI system,
632 there is a possibility of it presenting false negatives.
633 Editors who rely solely on this system could ad-
634 versely impact the review process. As an AI-based
635 model, this tool is prone to errors. Therefore, ed-
636 itors and chairs are advised to use this tool only
637 for assistance, to verify the results, and to analyze
638 them carefully before making any decisions.

⁸<https://openai.com/index/introducing-chatgpt-enterprise/>

⁹<https://arxiv.org/>

We hope that our data and analyses will facilitate
constructive discussions within the community and
help prevent the misuse of AI.

References

- Sahar Abdelnabi and Mario Fritz. 2021. *Adversarial watermarking transformer: Towards tracing text provenance with data hiding*. In *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*, pages 121–140. IEEE.
- OpenAI Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Benjamin Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Sim'on Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Raphael Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Lukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Ryan Kiros, Matthew Knight, Daniel Kokotajlo, Lukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kopic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Adeola Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel P.

698	Mossing, Tong Mu, Mira Murati, Oleg Murk, David M'ely, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Ouyang Long, Cullen O'Keefe, Jakub W. Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alexandre Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Pondé de Oliveira Pinto, Michael Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario D. Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Shepard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin D. Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas A. Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cer'on Uribe, Andrea Valone, Arun Vijayvergiya, Chelsea Voss, Carroll L. Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2023. Gpt-4 technical report .	
736	Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2023a. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature . <i>CoRR</i> , abs/2310.05130.	
741	Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2023b. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature . <i>arXiv preprint arXiv:2310.05130</i> .	
746	Saeed Awadh Bin-Nashwan, Mouad Sadallah, and Mohamed Bouteraa. 2023. Use of chatgpt in academia: Academic integrity hangs in the balance . <i>Technology in Society</i> , 75:102370.	
750	Megha Chakraborty, SM Tonmoy, SM Zaman, Krish Sharma, Niyar R Barman, Chandan Gupta, Shreya Gautam, Tanay Kumar, Vinija Jain, Aman Chadha, et al. 2023a. Counter turing test ct^2: Ai-generated text detection is not as easy as you may think—introducing ai detectability index . <i>arXiv preprint arXiv:2310.05030</i> .	
	Souradip Chakraborty, Amrit Singh Bedi, Sicheng Zhu, Bang An, Dinesh Manocha, and Furong Huang. 2023b. On the possibilities of ai-generated text detection . <i>CoRR</i> , abs/2304.04736.	757 758 759 760
	Yutian Chen, Hao Kang, Vivian Zhai, Liangze Li, Rita Singh, and Bhiksha Raj. 2023. Gpt-sentinel: Distinguishing human and chatgpt generated content . <i>CoRR</i> , abs/2305.07969.	761 762 763 764
	Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways . <i>Journal of Machine Learning Research</i> , 24(240):1–113.	765 766 767 768 769 770
	The Editor Engagement Chris Graf. 2022. Upholding research integrity and publishing ethics – identifying ethical concerns .	771 772 773
	Alessandro Gambetti and Qiwei Han. 2024. Aigenfoodreview: A multimodal dataset of machine-generated restaurant reviews and images on social media . <i>arXiv preprint arXiv:2401.08825</i> .	774 775 776 777
	Sebastian Gehrmann, Hendrik Strobelt, and Alexander M. Rush. 2019. GLTR: statistical detection and visualization of generated text . In <i>Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations</i> , pages 111–116. Association for Computational Linguistics.	778 779 780 781 782 783 784 785
	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 . <i>CoRR</i> , abs/2301.07597.	786 787 788 789 790
	Beatriz Gutiérrez-Caneda, Jorge Vázquez-Herrero, and Xosé López-García. 2023. Ai application in journalism: Chatgpt and the uses and risks of an emergent technology . <i>Profesional de la información</i> , 32(5).	791 792 793 794
	Veronika Hackl, Alexandra Elena Müller, Michael Granitzer, and Maximilian Sailer. 2023. Is GPT-4 a reliable rater? evaluating consistency in GPT-4 text ratings . <i>CoRR</i> , abs/2308.02575.	795 796 797 798
	Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. RADAR: robust ai-text detection via adversarial learning . In <i>Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023</i> .	799 800 801 802 803 804
	Oana Ignat, Xiaomeng Xu, and Rada Mihalcea. 2024. Maide-up: Multilingual deception detection of gpt-generated hotel reviews . <i>arXiv preprint arXiv:2404.12938</i> .	805 806 807 808
	Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023. Towards mitigating LLM hallucination via self reflection . In <i>Findings</i>	809 810 811

812	<i>of the Association for Computational Linguistics: EMNLP 2023</i> , pages 1827–1843, Singapore. Association for Computational Linguistics.	868
813		869
814		870
815	Yury Kashnitsky, Drahomira Herrmannova, Anita de Waard, Georgios Tsatsaronis, Catriona Fennell, and Cyril Labbé. 2022. Overview of the dagpap22 shared task on detecting automatically generated scientific papers. In <i>Third Workshop on Scholarly Document Processing</i> .	871
816		872
817		873
818		874
819		875
820		876
821	John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models . In <i>International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA</i> , volume 202 of <i>Proceedings of Machine Learning Research</i> , pages 17061–17084. PMLR.	877
822		878
823		879
824		880
825		881
826		882
827		883
828	Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2024. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. <i>Advances in Neural Information Processing Systems</i> , 36.	884
829		885
830		886
831		887
832		888
833	Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023. Deepfake text detection in the wild. <i>arXiv preprint arXiv:2305.13242</i> .	889
834		890
835		891
836		892
837	Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Hao-tian Ye, Sheng Liu, Zhi Huang, Daniel A. McFarland, and James Y. Zou. 2024a. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on AI conference peer reviews . <i>CoRR</i> , abs/2403.07183.	893
838		894
839		895
840		896
841		897
842		898
843		899
844	Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Hao-tian Ye, Sheng Liu, Zhi Huang, Daniel A. McFarland, and James Y. Zou. 2024b. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on ai conference peer reviews . <i>ArXiv</i> , abs/2403.07183.	900
845		901
846		902
847		903
848		904
849		905
850		906
851	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 . <i>CoRR</i> , abs/1907.11692.	907
852		908
853		909
854		910
855		911
856	Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature . In <i>International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA</i> , volume 202 of <i>Proceedings of Machine Learning Research</i> , pages 24950–24962. PMLR.	912
857		913
858		914
859		915
860		916
861		917
862		918
863		919
864	Travis J. E. Munyer and Xin Zhong. 2023. Deep-textmark: Deep learning based text watermarking for detection of large language model generated text . <i>CoRR</i> , abs/2305.05773.	920
865		921
866		922
867		923
		924
	Yikang Pan, Liangming Pan, Wenhui Chen, Preslav Nakov, Min-Yen Kan, and William Yang Wang. 2023. On the risk of misinformation pollution with large language models . In <i>Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023</i> , pages 1389–1403. Association for Computational Linguistics.	
	Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library . In <i>Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada</i> , pages 8024–8035.	
	Jipeng Qiang, Shiyu Zhu, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. 2023. Natural language watermarking via paraphraser-based lexical substitution . <i>Artif. Intell.</i> , 317:103859.	
	Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer . <i>J. Mach. Learn. Res.</i> , 21:140:1–140:67.	
	Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, S.M Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models - an extensive definition, quantification, and prescriptive remediations . In <i>Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing</i> , pages 2541–2573, Singapore. Association for Computational Linguistics.	
	Domenic Rosati. 2022. Synscipass: detecting appropriate uses of scientific text generation. <i>arXiv preprint arXiv:2209.03742</i> .	
	Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2023a. Can ai-generated text be reliably detected? <i>CoRR</i> , abs/2303.11156.	
	Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2023b. Can ai-generated text be reliably detected? <i>arXiv preprint arXiv:2303.11156</i> .	
	Wajiha Shahid, Yiran Li, Dakota Staples, Gulshan Amin Gilkar, Saqib Hakak, and Ali A. Ghorbani. 2022. Are you a cyborg, bot or human? - A survey on detecting fake news spreaders . <i>IEEE Access</i> , 10:27069–27083.	
	Xiaoming Shi, Zeming Liu, Chuan Wang, Haitao Leng, Kui Xue, Xiaofan Zhang, and Shaoting Zhang. 2023. Midmed: Towards mixed-type dialogues for medical consultation . <i>arXiv preprint arXiv:2306.02923</i> .	

925	Yi Shi and Lin Sun. 2024. How generative ai is transforming journalism: Development, application and ethics . <i>Journalism and Media</i> , 5(2):582–594.	980
926		981
927		
928	Ergon Cugler de Moraes Silva and Jose Carlos Vaz. 2024. How disinformation and fake news impact public policies?: A review of international literature . <i>arXiv preprint arXiv:2406.00951</i> .	982
929		983
930		984
931		985
932	Karanpartap Singh and James Zou. 2023. New evaluation metrics capture quality degradation due to LLM watermarking . <i>CoRR</i> , abs/2312.02382.	986
933		987
934		988
935	Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askeil, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models . <i>CoRR</i> , abs/1908.09203.	989
936		990
937		991
938		
939		
940	Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models . <i>arXiv preprint arXiv:2312.11805</i> .	992
941		993
942		994
943		995
944		
945		
946	Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Sergey I. Nikolenko, Evgeny Burnaev, Serguei Barannikov, and Irina Piontkovskaya. 2023. Intrinsic dimension estimation for robust detection of ai-generated texts . In <i>Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023</i> .	996
947		997
948		998
949		999
950		1000
951		1001
952		1002
953		1003
954		1004
955	Debora Weber-Wulff, Alla Anohina-Naumecca, Sonja Bjelobaba, Tomas Foltynek, Jean Guerrero-Dib, Olu-mide Popoola, Petr Sigut, and Lorna Waddington. 2023. Testing of detection tools for ai-generated text . <i>CoRR</i> , abs/2306.15666.	
956		
957		
958		
959		
960	Kyle Wiggers. 2022. Openai’s attempts to watermark ai text hit limits . <i>TechCrunch</i> , December, 10.	
961		
962	Max Wolff. 2020. Attacking neural text detectors . <i>CoRR</i> , abs/2002.11768.	
963		
964	Kangxi Wu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2023. Llmdet: A third party large language models generated text detection tool . In <i>Findings of the Association for Computational Linguistics: EMNLP 2023</i> , pages 2113–2133.	
965		
966		
967		
968		
969	KiYoon Yoo, Wonhyuk Ahn, Jiho Jang, and Nojun Kwak. 2023. Robust multi-bit natural language watermarking through invariant features . In <i>Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023</i> , pages 2092–2115. Association for Computational Linguistics.	
970		
971		
972		
973		
974		
975		
976		
977	Xiao Yu, Yuang Qi, Kejiang Chen, Guoqiang Chen, Xi Yang, Pengyuan Zhu, Weiming Zhang, and Neng-hai Yu. 2023. GPT paternity test: GPT generated text detection with GPT genetic inheritance . <i>CoRR</i> , abs/2305.12519.	980
978		981
979		
	Xuan Zhang and Wei Gao. 2023. Towards llm-based fact verification on news claims with a hierarchical step-by-step prompting method . In <i>Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, IJCNLP 2023 -Volume 1: Long Papers, Nusa Dua, Bali, November 1 - 4, 2023</i> , pages 996–1011. Association for Computational Linguistics.	982
		983
		984
		985
		986
		987
		988
		989
		990
		991
	Xuandong Zhao, Yu-Xiang Wang, and Lei Li. 2023. Protecting language generation models via invisible watermarking . <i>CoRR</i> , abs/2302.03162.	992
		993
		994
	A Dataset	995
	We generated a fake review for each paper using both GPT-3.5 and GPT-4. We gave the prompt template similar to the both conference style of reviews. We also generated regenerated reviews for this task.	996
		997
		998
		999
		1000
	We discuss the dataset in more details in Appendix Section	1001
		1002
	Below is the prompt we used for generating AI-generated review ICLR 2022 reviews:	1003
		1004
	<p>System: You are a research scientist reviewing a scientific paper.</p> <p>User: Read the following paper and write a thorough peer-review in the following format:</p> <ol style="list-style-type: none"> 1) Summary of the paper 2) Main review 3) Summary of the review <p style="text-align: center;">[paper text]</p>	1005
	Below is the prompt we used for generating AI-generated review NeurIPS 2022 reviews:	1006
		1007
	<p>System: You are a research scientist reviewing a scientific paper.</p> <p>User: Read the following paper and write a thorough peer-review in the following format:</p> <ol style="list-style-type: none"> 1) Summary (avg word length 100) 2) Strengths and weaknesses 3) Questions 4) Limitations (in short) <p style="text-align: center;">[paper text]</p>	1008

1009 Below is the prompt we used for generating AI
1010 regenerated review ICLR 2022 reviews :-

System: You are a research scientist reviewing a scientific paper.

User: Your task is to draft a high-quality peer-review in the below format:

- 1) Summarize the paper.
- 2) List strong and weak points of the paper, Question and Feedback to the author. Be as comprehensive as possible.
- 3) Write review summary (Provide supporting arguments for your recommendation).

[paper text]

1011
1012 To generate AI-regenerated reviews, we used
1013 prompts that were very distinct from those we used
1014 to generate AI reviews for training. The reason for
1015 this approach is that a reviewer may write any kind
1016 of prompt, which could be very different from the
1017 prompts we used for training.

1018 Below is the prompt we used for generating AI
1019 regenerated review NeurIPS 2022 reviews :-

System: You are a research scientist reviewing a scientific paper.

User: Your task is to draft a high-quality peer-review in the below format:

- 1) Briefly summarize the paper and its contributions
- 2) Please provide a thorough assessment of the strengths and weaknesses of the paper
- 3) Please list up and carefully describe any questions and suggestions for the authors
- 4) Limitations: Have the authors adequately addressed the limitations and potential negative societal impact of their work? If not, please include constructive suggestions for improvement. Write in few lines only

[paper text]

1020 1021 **B Error Analysis**

1022 We conducted an analysis of the predictions made
1023 by our proposed baseline to identify the areas
1024 where it most frequently fails.

1025 **B.1 Challenges after paraphrasing:**

1026 Our regeneration-based approach sometimes fails
1027 when it processes a paraphrased review. Paraphras-

ing can alter the semantics of a review to some extent, leading to discrepancies with our reverse-generated reviews. Consequently, our model may incorrectly predict these as human-written rather than AI-generated. Our proposed defense strategy corrects only the tokens that have been changed during paraphrasing. However, when the paraphrasing significantly alters the style, our RR model fails.

1036 **B.2 Sometimes Regenerated review and AI 1037 written reviews are similar:**

1038 Our RR model work on the similarity of review
1039 and Regenerated review. We found the model fails
1040 when LLM generates a review that is very much
1041 similar to the human writing. In those cases we
1042 found that the similarity score will be low and
1043 hence the model fails.

1044 Our RR model focuses on the similarity between
1045 the original review and the regenerated review. We
1046 discovered that the model struggles when a AI-
1047 generated a review closely resembling regenerated
1048 review but is human written. In such instances,
1049 the similarity score tends to be low, leading to the
1050 model's failure. This suggests that the model may
1051 not effectively differentiate between human-like
1052 text generated by AI.

1053 **C Token Attack**

1054 Below is an example of how impactful various at-
1055 tacks can be when replacing words in a review :-
1056 After reviewing all the attacks, we observe that
1057 the adjective attack produced more logical changes
1058 compared to the others. For example, in the noun
1059 attack, 'model' was replaced with 'pose,' 'learning'
1060 with 'discovery,' 'performance' with 'execution,'
1061 and 'datasets' with 'information sets,' which are
1062 not very meaningful and thus make the attack less
1063 effective. Replacing words can cause significant
1064 changes in the meaning of a review and can even
1065 alter the context. So we used only the adjective
1066 attack for our experiments.

Actual Sentence: The **model** is evaluated in both reinforcement **learning** and vision settings, **showcasing significant performance** boosts in **tasks** such as DMC Suite with distractors and CIFAR-10/STL10 **datasets**.

Adjective: The model is evaluated in both reinforcement learning and vision settings, showcasing **substantial** performance boosts in tasks such as DMC Suite with distractors and CIFAR-10/STL-10 datasets.

Noun: The **pose** is evaluated in both reinforcement **discover** and vision scene, showcasing significant **execution** boosts in project such as DMC Suite with distractors and CIFAR-10/STL-10 **informationsets**.

Adverb: The model is evaluated in both reinforcement learning and vision settings, **showce-quallyng** significant performance boosts in **tequallyks** such **equally** DMC Suite with distractors and CIFAR-10/STL-10 **datequalhowevers**

D Baseline Comparison

D.1 RADAR (Hu et al., 2023)

The way RADAR works is as follows - First an AI-text corpus is generated from a target (frozen) language model from a human-text corpus. The next step is followed by introduction of a paraphraser (a tunable language model) and a detector (a separate tunable language model). In the training stage, the detector’s objective is to distinguish between human-generated text and AI-generated text, whereas the paraphraser’s goal is to rephrase AI-generated text to avoid detection. The model parameters of the paraphraser and detector are updated in an adversarial learning manner. During the evaluation (testing) phase, the deployed detector utilizes its training to assess the probability of content being AI-generated for any given input instance.

D.2 LLMDet (Wu et al., 2023):

The overall framework of the system consists of two main components - 1) Dictionary creation and 2) Text detection. The main idea was to make use of the perplexity as a measurement of identifying the generated text from different LLMs. So the dictionary had n -grams as keys and the next to-

ken probabilities as values. The dictionary serves as prior information during the detection process. Since the dictionary of n -grams and their probabilities was obtained it enabled the utilization of the corresponding dictionary of each model as prior information for third-party detection, facilitating the calculation of the proxy perplexity of the text being detected on each model. Proxy perplexity was then used as a feature into a trained text classifier, the corresponding detection results were obtained.

D.3 DEEP-FAKE (Li et al., 2023)

To determine whether machine-generated text can be discerned from human-written content, the collected data was categorized into six settings based on the sources used for model training and evaluation. These settings progressively increased difficulty for detection. The classifier then assigned a probability to each text, indicating the likelihood of it being authored by humans or generated by language model models (LLMs). AvgRec (average recall) was the principal metric, calculated as the average score between the recall on human-written texts (HumanRec) and the recall on machine-generated texts (MachineRec).

D.4 FAST-DETECT GPT (Bao et al., 2023b)

The model comprises of a three-fold architecture - 1) Revealing and confirming a novel conjecture that humans and machines exhibit distinct word selection patterns within a given context. 2) Employing conditional probability curvature as a fresh characteristic to identify machine-generated text, thereby reducing the detection expenses by a factor of 2 orders of magnitude. 3) Attaining the highest average detection accuracy in both white-box and black-box environments and comparing to current zero-shot text detection systems.

E Paraphrasing

We performed paraphrasing by providing prompts to the Gemini model (Team et al., 2023). We have provided example of paraphrased review in table 6

We used the following prompt for generating paraphrased text:-

System: You are a paraphraser.
User: Paraphrase the following review:

[Review]

Actual Review	
1. Summary of the Paper:-	The paper explores the incorporation of higher-order dynamics specifically second derivatives into neural models to improve the estimation of cardiac pulse dynamics. The focus is on video-based vital sign measurement particularly Photoplethysmography (PPG) using deep learning architectures. The research demonstrates that optimizing for second derivatives in the loss function enhances the estimation of waveform morphology crucial for clinically significant scenarios such as left ventricle ejection time (LVET) intervals. The study uses simulation-generated data for training due to the scarcity of labeled real data and evaluates model performance against a real dataset.
2. Main Review:-	The paper presents a novel approach by considering higher-order dynamics in the context of video-based cardiac measurements a crucial step towards capturing subtle variations in arterial health indicators. The methodology is well-structured building on existing literature and providing a clear rationale for exploring multiderivative learning objectives in neural models. The experiments are detailed utilizing synthetic data for training and real data for evaluation demonstrating the feasibility and effectiveness of incorporating second derivatives in deep learning models for cardiac pulse estimation. The systematic evaluation conducted by the authors is rigorous focusing on quantitative metrics such as heart rate (HR) and left ventricle ejection time (LVET) intervals to compare different model configurations. The results indicate improvements in LVET estimation when second derivative frames are included highlighting the benefits of considering higher-order dynamics in the estimation process. The study also includes a qualitative analysis providing visual representations of the predicted signals and comparing them to ground truth data enhancing the interpretation of the quantitative results. Overall the inclusion of both quantitative and qualitative assessments strengthens the paper's conclusions and provides a comprehensive understanding of the proposed methodology.
3. Summary of the Review:-	The paper makes a valuable contribution to the field of video-based cardiac measurement by emphasizing the importance of higher-order dynamics in accurately estimating vital sign parameters. The research framework is well-built supported by a thorough literature review detailed experiments and insightful conclusions. The results demonstrate the significance of optimizing for second derivatives in neural models improving the estimation of clinically relevant waveform morphology especially in scenarios like LVET intervals. Overall the paper presents a compelling case for incorporating multiderivative learning objectives to enhance the accuracy of video-based vital sign measurements.
Paraphrased Review	
1. Summary of the Paper:	The paper investigates including second derivatives (higher-order dynamics) in neural models to accurately measure cardiac pulse signals using video (Photoplethysmography or PPG). By focusing on these higher-order dynamics, the models can better capture subtle changes in these signals, leading to improved estimates of waveform characteristics. Since real-world data is limited, the models are trained on simulated data and then tested on real data.
2. Main Review:	This novel approach is a step forward in understanding subtle fluctuations in cardiac health indicators through video-based measurements. The paper builds on existing research, providing a strong rationale for incorporating higher-order dynamics into neural models for cardiac pulse estimation. The experiments use both simulated and real data, demonstrating the effectiveness of including second derivatives in these models. Rigorous evaluation using metrics like heart rate and LVET intervals shows that incorporating second derivatives improves LVET estimation. Qualitative analysis, comparing predicted signals to ground truth, further supports these findings.
3. Summary of the Review:	This paper significantly contributes to video-based cardiac measurements by highlighting the importance of higher-order dynamics in accurately estimating vital sign parameters. The well-structured research framework, detailed experiments, and insightful conclusions demonstrate the value of optimizing for second derivatives in neural models. This approach enhances waveform morphology estimation, especially for clinically important measures like LVET intervals, making it a valuable addition to the field.

Table 4: Examples of Actual and Paraphrased Review

Model	Precision		Recall		F1 - Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
RADAR	29.58	31.75	79.60	93.05	69.29	70.72	60.12	62.37
LLMDT	19.38	18.64	98.03	98.61	32.36	31.35	22.13	21.46
DEEP-FAKE	76.68	75.81	97.37	0.9792	85.80	85.45	86.35	86.32
FAST DETECT	84.88	82.31	96.05	84.03	90.12	83.16	96.00	93.81
Our RR Model	99.34	95.14	93.79	92.57	96.49	93.84	98.49	97.36

Table 5: Comparison Result of proposed Review Regeneration technique; Here the AI-generated reviews and regenerated reviews are generated by GPT-3.5. ; RR: Review Regeneration; TF: Token Frequency

Model	Precision		Recall		F1-Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
ADJECTIVE	99.99	99.99	99.80	99.30	99.99	99.65	99.92	99.82
NOUN	91.45	99.99	99.99	99.99	95.53	99.99	98.50	99.99
ADVERB	93.42	90.97	89.86	90.35	91.61	90.66	97.00	95.16

Table 6: Result of Token Frequency based Approach. Here the fake review is generated by prompting GPT-4

Model	Precision		Recall		F1-Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
ADJECTIVE	99.99	99.99	98.70	99.32	99.35	99.66	99.77	99.82
NOUN	98.69	99.99	99.34	97.92	99.02	98.95	99.65	99.46
ADVERB	96.55	97.24	92.11	97.92	94.28	97.58	98.03	98.75

Table 7: Result of Token Frequency-based Approach. Here the fake review is generated by prompting GPT-3.5

Model	Precision		Recall		F1-Score		Accuracy	
	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS	ICLR	NeurIPS
RADAR	14.47	10.42	59.46	57.69	23.28	17.65	52.30	51.39
LLMDET	97.37	95.77	50.68	49.64	66.67	65.38	51.32	50.00
DEEP-FAKE	35.38	44.44	71.88	59.26	47.42	50.79	55.91	56.94
FAST DETECT	5.26	7.64	80.00	84.62	9.88	14.01	67.84	68.31
Our TF Model	76.92	64.29	74.19	84.38	75.53	72.97	95.40	93.73
Our RR Model	90.87	93.98	78.62	81.25	84.30	87.15	91.51	92.86

Table 8: Comparison results after paraphrasing applying Paraphrasing defence