

CoCoNUTS: Concentrating on Content while Neglecting Uninformative Textual Styles for AI-Generated Peer Review Detection

Yihan Chen^{1,2}, Jiawei Chen^{1,2}, Guozhao Mo^{1,2}, Xuanang Chen²,
Ben He^{1,2*}, Xianpei Han², Le Sun²

¹University of Chinese Academy of Sciences, Beijing, China

²Chinese Information Processing Laboratory, Institute of Software,
Chinese Academy of Sciences, Beijing, China
chenyihan20241@iscas.ac.cn, benhe@ucas.ac.cn

Abstract

The growing use of large language models (LLMs) in peer review threatens scholarly integrity. Recent conference policies allow AI tools for language polishing but prohibit their use for generating substantive content. However, existing detectors mainly rely on stylistic cues, making it difficult to distinguish between surface-level language refinement and genuine content generation. To address this, we advocate a content-based detection paradigm and introduce CoCoNUTS, a comprehensive benchmark containing 315,535 reviews covering leading AI conferences and six human-AI collaboration modes. Our evaluation shows that current detectors struggle to handle these nuanced settings. Consequently, we propose CoCoDet, an AI review detector designed to identify substantive AI-generation. Experiments demonstrate that CoCoDet achieves a macro F1-score of 98.24%. Crucially, on permissible machine-polished reviews, it maintains a low false positive rate of 3.89%, substantially outperforming the strongest baseline (7.84%). Examination on real-world reviews using CoCoDet reveals an escalating trend of substantive AI generation. Our work exposes the inadequacy of current detectors, underscoring the importance of domain-specific solutions. Our code is available at <https://github.com/icip-cas/CoCoNUTS>.

1 Introduction

Peer review is an essential part of academic publications. However, the rapid advancement of Large Language Models (LLMs) has raised significant concerns about the misuse of LLMs in peer review. These concerns are reflected in official policies of academic conferences. For instance, policies from venues like ACL, ICML, and ICLR permit using LLMs for language polishing but strictly prohibit using them for generating substantive content (ACL Rolling Review; ICML; ICLR). Although these

policies aim to regulate the use of LLMs, a recent study indicates a rising trend in the use of LLMs for generating substantial review content (Liang et al., 2024). This misuse not only creates a risk of data leakage but also results in unreliable reviews. Research indicates that LLMs struggle to adequately evaluate scientific contributions (Ye et al., 2024), generating hollow reviews (Zhou et al., 2024a; Du et al., 2024), and are vulnerable to manipulation, demonstrating an urgent need for accurate classification and detection of AI usage in peer review.

Current general-purpose AI-generated text detectors face a dual challenge. They are vulnerable to paraphrasing attacks, allowing humanized AI-generated content to go undetected (Sadasivan et al., 2025; Zhou et al., 2024b). Simultaneously, they exhibit high false-positive rates on text with even minor AI polishing, unjustly flagging permissible use (Saha and Feizi, 2025). Given that paraphrasing is a semantics-invariant operation (Su et al., 2024) that transfers textual style without altering content, these failures suggest a focus on textual style while neglecting content. Although textual style is a distinguishable feature of AI writing (Krishna et al., 2023), this stylistic dependency is problematic in the peer review context, as it risks both unjustly penalizing legitimate AI assistance and overlooking deceptively generated reviews.

To this end, we advocate a paradigm shift in AI-generated review detection by emphasizing content composition over textual style. First, we introduce CoCoNUTS, a comprehensive benchmark constructed based on real-world reviews from leading AI conferences and features six meticulously simulated human-AI collaboration modes, ranging from pure human reviews to fully AI-generated reviews. These reviews are categorized into three classes based on content composition: Human, Mix, and AI. Second, we propose CoCoDet, a content-concentrated AI review detector. To disentangle content features from stylistic

*Corresponding author.

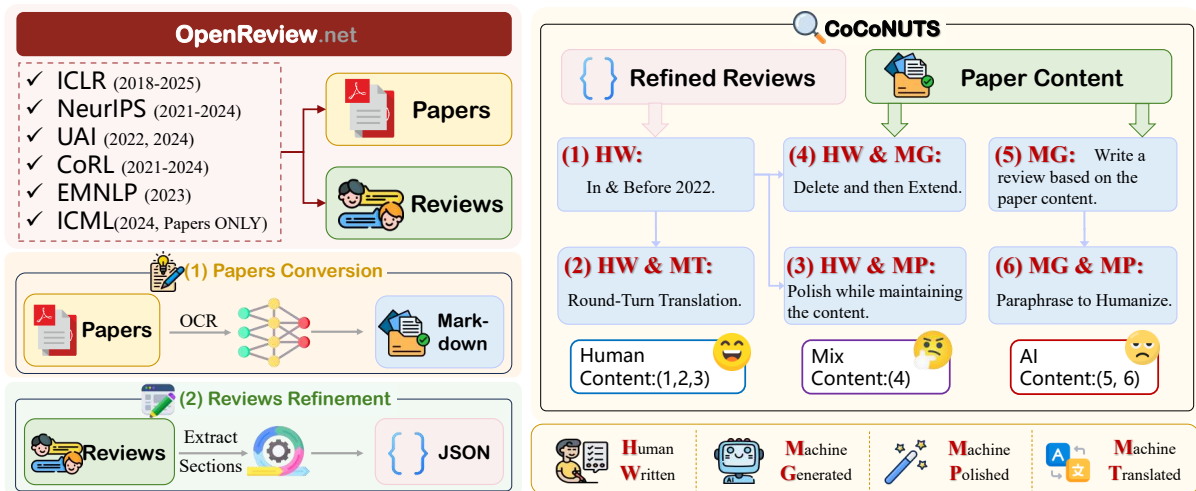


Figure 1: Overview of our CoCoNUTS benchmark. The left side illustrates data acquisition and preprocessing, while the right side shows the construction pipeline and the detection task based on content composition.

tic cues, CoCoDet is trained with a multi-task framework, comprising a primary content composition identification task and three auxiliary tasks. Together, CoCoNUTS and CoCoDet establish a content-centric framework for AI review detection.

Building on this, we conduct a comprehensive evaluation of a wide range of AI-generated text detectors on the CoCoNUTS benchmark. Our results reveal that LLM-based detectors, even with few-shot prompting, struggle to focus on substantive content and tend to rely on stylistic cues, leading to unreliable predictions. Similarly, general-purpose detectors (encompassing both metric-based and fine-tuned models) perform poorly in this content-based task, especially non-style-robust models, which fail to produce reliable predictions. CoCoDet achieves state-of-the-art performance, with a macro F1-score exceeding 98% on the ternary detection task, substantially outperforming baseline detectors. Furthermore, applying CoCoDet to real-world reviews reveals a clear year-over-year increase in AI usage, spanning both AI-assisted polishing and purely AI-generated reviews. This trend underscores the urgency of adopting robust, content-based detection methods.

Our contributions are tri-fold: (1) We introduce CoCoNUTS, a comprehensive benchmark derived from 315,535 reviews. By simulating complex real world human-AI collaboration modes, it provides fine-grained resources for the AI review detection. (2) We conduct a comprehensive evaluation of 9 baselines, finding that out-of-box LLMs are ill-suited for this task, and general-purpose detectors exhibit significant performance dispar-

ities correlated with style-robustness and reveal their overall inadequacy, underscoring the critical importance of domain-specific solutions. (3) We propose CoCoDet, a content-centric AI review detector. Our evaluation on pre-ChatGPT and AI-polished reviews confirms its reliability. Building on this, we analyze recent real-world reviews and reveal an escalating trend of supposedly prohibited AI usage.

2 CoCoNUTS Benchmark

To facilitate a fair and robust evaluation of LLM involvement in academic peer review, we introduce CoCoNUTS. The overall dataset construction and detection tasks are illustrated in Figure 1.

2.1 Dataset Construction

To address the limitations of existing datasets in representing diverse AI usage in peer review, we construct a large-scale dataset of 315,535 instances, comprising six categories designed to simulate realistic human-AI collaboration modes. The detailed construction process is as follows.

First, we collect reviews and corresponding manuscripts from OpenReview, covering venues including ICLR (2018–2025), NeurIPS (2021–2024), UAI (2022, 2024), CoRL (2021–2024), and EMNLP (2023). To enhance the topical diversity of generated reviews, we incorporate papers from ICML 2024. From the collected reviews, we extract the substantive parts, such as the analysis and questions, while discarding templated content like ratings and confidence scores. This step refines the data by eliminating variations from different review

Benchmark	Size	Category						Attribution	
		HW	MG	HWMG	HWMT	HWMP	MGMP	Coupled	Separate
General									
MGTBench (He et al., 2024)	2.82k	✓	✓	✗	✗	✗	✓	✓	✗
M4 (Wang et al., 2024b)	122k	✓	✓	✗	✗	✓	✗	✓	✗
HC3 Plus (Su et al., 2024)	210k	✓	✗	✗	✓	✓	✗	✗	✗
RAID (Dugan et al., 2024)	6.2M	✓	✓	✗	✗	✗	✓	✓	✗
MixSet (Zhang et al., 2024)	3.6k	✓	✓	✓	✗	✓	✓	✓	✗
M4GT-Bench (Wang et al., 2024a)	217k	✓	✓	✓	✗	✗	✗	✓	✗
CUDRT (Tao et al., 2024)	480k	✓	✓	✓	✓	✓	✗	✓	✗
LAMP (Chakrabarty et al., 2025)	1.06k	✓	✓	✓	✗	✗	✓	✓	✗
Beemo (Artemova et al., 2025)	19.6k	✓	✓	✓	✗	✗	✓	✓	✗
HART (Bao et al., 2025)	32k	✓	✓	✗	✗	✓	✓	✗	✗
OpenTuringBench (Cava and Tagarelli, 2025)	543k	✓	✓	✓	✗	✓	✓	✓	✗
News									
TuringBench (Uchendu et al., 2021)	200k	✓	✓	✗	✗	✗	✗	✓	✗
LLMDetect (Cheng et al., 2025)	64.3k	✓	✓	✓	✗	✓	✗	✓	✗
Academic									
FAIDset (Ta et al., 2025)	83.3k	✓	✓	✓	✗	✓	✓	✓	✗
AIPR-Detection-Benchmark (Yu et al., 2025)	789k	✓	✓	✗	✗	✗	✗	✓	✗
CoCoNUTS (Ours)	316k	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison of publicly available resources for AI-Generated Text detection. Our work is the only resource that includes texts from six distinct usage scenarios. Furthermore, we are the first to propose disentangled model attribution, with separate labels for the initial content generator and the final style modifier.

templates. Concurrently, all collected papers are converted from PDF to Markdown format using Nougat (Blecher et al., 2024) to facilitate further processing.

Next, we construct the six categories using a carefully designed generation pipeline. This pipeline employs various LLMs, including DeepSeek-R1 (Guo et al., 2025), Gemini-2.5-flash (Comanici et al., 2025), Llama-3.3-70B-Instruct (Dubey et al., 2024), Qwen2.5-72B-Instruct (Team, 2024), and Qwen3-32B (Team, 2025). The construction methods are as follows (specific prompts are detailed in Appendix C):

- **HW** (Human-Written): To ensure a rigorous and uncontaminated baseline, we exclusively select reviews submitted in or before 2022 (prior to the public release of ChatGPT). The detailed data scope is further elucidated in Appendix C.3.
- **HWMT** (Human-Written & Machine-Translated): We apply back-translation to HW reviews using LLMs, translating them into Chinese and then back into English.
- **HWMP** (Human-Written & Machine-Polished): We prompt LLMs to polish HW reviews, enhancing sentence structure while preserving the original content.

- **HWMG** (Human-Written & Machine-Generated): We provide LLMs with both original HW reviews and corresponding paper sections. The model is prompted to prune redundant parts from the HW review and add missing critical points (e.g., weaknesses).
- **MG** (Machine-Generated): We employ a few-shot prompting strategy, providing the LLMs with several HW reviews, followed by the content of a source paper, tasking the LLMs to generate reviews.
- **MGMP** (Machine-Generated & Machine-Paraphrased): We process the MG reviews with a different LLM prompted to paraphrase, simulating a reviewer trying to humanize the content to evade AI detection.

Finally, a filtering pipeline is employed to ensure data quality. First, we eliminate conversational artifacts (e.g., “Here is the polished review”). To mitigate length bias, we then filter outliers by only retaining samples within the 5th-95th percentile length range of the HW set. Quality is validated through manual inspection of 100 random instances. For the MG category, we further incorporate reviews generated by Claude 3.5 Sonnet and GPT-4o from the AIPR benchmark (Yu et al., 2025) to enhance diversity. The final statistics are

in Appendix C.2.

2.2 Task Definition

The task of AI text detection faces a fundamental trade-off between expressiveness and robustness. Binary detection (Guo et al., 2023) lacks the expressiveness for the scenarios of human-AI co-authorship, while fine-grained paradigms, such as model-specific attribution (Ta et al., 2025) are vulnerable against unseen models and collaboration patterns. To bridge this gap, we introduce a ternary classification based on the content composition. We posit that semantic-invariant operations alter textual style without changing substantive content (see Appendix E.7 for analysis). Based on this principle, we map our six categories into three content-based classes. Specifically, reviews with purely human content, even after such operations (HW, HWMT, HWMP), are labeled as “Human”. Conversely, reviews with purely AI content, including paraphrased versions (MG, MGMP), are labeled as “AI”. The hybrid scenario (HWMG) with content from both human and AI, is designated as “Mix”. This ternary setup serves as the basis for our detection task and evaluation.

3 CoCoDet Detector

As mentioned above, current detectors struggle to distinguish between AI polishing and prohibited substantive generation in peer reviews. To address this domain-specific challenge, we introduce the Content-Concentrated Detector (CoCoDet) with a tailored multi-task training framework. To enable the model to learn robust representations, this framework integrates a primary task, *Content Composition Identification* from the CoCoNUTS benchmark, with three carefully designed auxiliary tasks: *Collaboration Mode Attribution*, *Content Source Attribution*, and *Textual Style Attribution*. These auxiliary tasks allow the model to separate content features from stylistic ones, enabling the precise identification of substantive AI generation.

Content Composition Identification The primary task requires identifying reviews into three classes based on their content composition: *Human*, *AI*, or *Mix*.

A key challenge in this task is to ensure clear decision boundaries. To this end, we first adopt the large margin cosine loss to enhance class separability (Wang et al., 2018). Let \mathbf{x} be the feature embedding and \mathbf{r}_j be the weight vector for class

j . The logit for class j is the cosine similarity $z_j = \cos(\theta_j)$, where θ_j is the angle between \mathbf{x} and \mathbf{r}_j . The base margin m_{base} is subtracted from the logit of the ground-truth class y :

$$z'_j = \begin{cases} z_j - m_{\text{base}} & \text{if } j = y \\ z_j & \text{if } j \neq y \end{cases} \quad (1)$$

Furthermore, considering that confusing *Human* and *AI* texts is more costly in our context, we introduce an additional cost margin, m_{cost} , to modify the logits of these critical negative classes during training. We formalize this entire mechanism as the *Cost-Sensitive Margin Loss (CSM-Loss)*. Based on the intermediate logits z'_j , we apply the cost margin m_{cost} to produce the final logits:

$$z_j^* = \begin{cases} s \cdot (z'_j + m_{\text{cost}}) & \text{if } j, y \in \{\text{human, ai}\} \text{ and } j \neq y \\ s \cdot z'_j & \text{otherwise} \end{cases} \quad (2)$$

where s is a scaling parameter. This targeted penalty structure compels the model to learn a more discriminative feature representation. Specifically, our CSM-Loss encourages inter-class separability, especially between the high-cost Human and AI classes. For example, a correct classification decision for a Human-class review requires that:

$$z_{\text{human}} > \max(z_{\text{ai}} + m_{\text{base}} + m_{\text{cost}}, z_{\text{mix}} + m_{\text{base}}) \quad (3)$$

Let C be the three classes. The loss $\mathcal{L}_{\text{main}}$ is the cross-entropy loss over these modified logits z_j^* :

$$\mathcal{L}_{\text{main}} = -\log \left(\frac{e^{z_y^*}}{\sum_{k=1}^C e^{z_k^*}} \right) \quad (4)$$

Content Source Attribution This multi-label classification task aims to trace the substantive content back to its origin by identifying the specific author that performed the initial generation. We operate on the hypothesis that human experts and different AI models possess unique features (e.g., knowledge cutoffs). These intrinsic differences lead to discernible, author-specific characteristics in the content they generate. By training the model to attribute content to its initial author, we compel it to move beyond superficial stylistic analysis and instead learn to identify who is capable of producing what kind of substantive critique. To capture these specific traits, we employ a direct, one-to-one mapping for the labels (e.g., “Qwen2.5” and “Qwen3” are treated as distinct labels). For a review first generated by Qwen3 and then polished by Gemini, the ground-truth label is “Qwen3”. For a Mix review

edited by Qwen3, the labels would be [“Human”, “Qwen3”] to reflect the dual contribution. The final loss \mathcal{L}_{con} is the binary cross-entropy (BCE) with logits:

$$\mathcal{L}_{\text{con}} = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\sigma(u_i)) + (1 - y_i) \log(1 - \sigma(u_i))] \quad (5)$$

where n is the number of content labels, y_i is the binary ground-truth, and u_i is the output logits from the final linear layer for content source attribution.

Textual Style Attribution This task seeks to identify the style modifier of a review in order to enable the detector to capture the stylistic patterns indicating the final authorship. In conjunction with the Content Source Attribution, this task enables the model to disentangle content from style (see Appendix B for details). In this task, based on the rationale that models from the same family develop consistent stylistic features, we group models into families (e.g., Qwen2.5 and Qwen3 are mapped to the “Qwen” label). For a review generated by Llama and then polished by Qwen2.5, the ground-truth label would be “Qwen”. This loss \mathcal{L}_{sty} is also the BCE with logits:

$$\mathcal{L}_{\text{sty}} = -\frac{1}{m} \sum_{i=1}^m [y_i \log(\sigma(v_i)) + (1 - y_i) \log(1 - \sigma(v_i))] \quad (6)$$

where m is the number of style labels, y_i is the binary ground-truth, and v_i is the output logits from the final linear layer.

Collaboration Mode Attribution This multi-class classification task enforces the model to understand the fine-grained compositional provenance of a text by attributing it to a specific collaboration mode. By classifying each review into one of the six predefined modes in the CoCoNUTS dataset, the model is explicitly informed about the latent hierarchy of content composition. The loss $\mathcal{L}_{\text{mode}}$ is the cross-entropy loss over logits:

$$\mathcal{L}_{\text{mode}} = -\log \left(\frac{e^{w_y}}{\sum_{j=1}^M e^{w_j}} \right) \quad (7)$$

where M is the total number of collaboration modes, w_y is the output logit from the final linear layer for the ground-truth class y , and w_j is the logit for the j -th class.

Based on these defined training tasks, we adopt ModernBERT (Warner et al., 2024) as the backbone of our CoCoDet detector. The model is

trained end-to-end using a composite loss function that linearly combines the loss from the primary task with weighted losses from the three auxiliary tasks. The composite loss \mathcal{L} is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \alpha \mathcal{L}_{\text{con}} + \beta \mathcal{L}_{\text{sty}} + \gamma \mathcal{L}_{\text{mode}} \quad (8)$$

where the weights α , β , and γ are hyper-parameters tuned on the validation set.

4 Experiments

We conduct a series of experiments on the CoCoNUTS benchmark to evaluate the CoCoDet detector. First, we compare CoCoDet with LLM-based detectors to show its effectiveness. To further reveal the limitations of general detectors, we evaluate their binary classification performance across the Human, Mix, and AI subsets of CoCoNUTS. We then perform an ablation study to validate the components of CoCoDet. Finally, we apply CoCoDet to analyze AI usage trends in real-world post-ChatGPT peer reviews.

4.1 Experimental Setup

Dataset We partition the dataset into training, validation, and test sets with an 8:1:1 ratio using stratified random sampling to ensure the distribution is consistent (see Appendix C.2 for details). All metrics are reported on the held-out test set.

Baselines We benchmark CoCoDet against nine baseline detection methods, including four recent LLMs and five mainstream general AI text detectors.

For the *LLM-based detectors*, we select DeepSeek-R1, Gemini-2.5-flash, Qwen2.5-72B-Instruct, and Qwen3-32B as baselines. Their performance is evaluated in both zero-shot and few-shot settings. Notably, for Gemini-2.5-Flash, we evaluate its performance in both its thinking and non-thinking modes. We prompt the models to classify the reviews by focusing on their substantive content composition rather than their stylistic features, and choose one from the three options: *Human*, *AI*, or *Mix*. In the few-shot scenario, we provide each model with one in-context example per class, ensuring the set of examples is identical for all models for a fair comparison.

For the *general detectors*, we select five mainstream methods, encompassing both model-based and metric-based approaches. The model-based methods include Radar (Hu et al., 2023) and LLM-DetectAIve (Abassy et al., 2024). The metric-based methods include LLMDet (Wu et al., 2023),

FastDetectGPT (Bao et al., 2024), and Binoculars (Hans et al., 2024). We strictly adhere to the officially recommended configurations for all general detectors. Model-based methods are run on the same hardware, while metric-based methods utilize their prescribed thresholds. For Binoculars, we report results at both accuracy and low-false-positive-rate (low-fpr) thresholds.

Training Details We fine-tune CoCoDet with the AdamW optimizer (Loshchilov and Hutter, 2019), selecting the best model on the validation set. Hyper-parameters are optimized via a sequential grid search (detailed in Appendix D.1).

Evaluation Metrics For LLM-based detectors, we report F1-scores for the Human, Mix, and AI classes, along with the average F1-score. For general detectors, we report the predicted AI rate on each class and the average accuracy, defined as the mean accuracy on the Human and AI classes. The detailed settings are provided in Appendix D.2. Additionally, to evaluate whether a detector relies on substantive content rather than stylistic features, we introduce the Style-Robustness metric. A detector is defined as style-robust if its Predicted AI Rate monotonically increases from the Human, to the Mix, and finally to the AI subset. To facilitate a fair comparison, we map the outputs of multi-class models. For LLM-DetectAIve, we map the “HW/HWMP” predictions to “Human” and “MG/MGMP” predictions to “AI”. For CoCoDet, we apply a more stringent standard: any non-Human prediction on Human subset is considered a false positive, while only an AI prediction is counted as a true positive on the AI subset.

4.2 Overall Results

Large language models are ill-suited for the task of content-concentrated detection. As detailed in Table 2, LLM-based detectors struggle to achieve reliable results on the CoCoNUTS benchmark. In a zero-shot setting, their performance is notably poor. Despite an improvement from few-shot prompting over a poor zero-shot baseline, the overall performance of LLM-based detectors remains severely limited. Even the best LLM fails to surpass a 60% average F1-score, while the majority of other models operate near or below chance levels. Furthermore, our additional experiments with increased examples (up to 8-shot) reveal that even with extensive guidance, the best-performing LLM still lags significantly be-

Detector	Human	Mix	AI	Average
LLMs (zero-shot)				
DeepSeek-R1	50.04	3.29	3.63	18.98
Gemini-2.5-flash(CoT)	56.01	2.81	47.87	35.56
Gemini-2.5-flash	57.28	12.37	49.80	39.82
Qwen2.5-72B-Instruct	48.47	3.05	16.82	22.78
Qwen3-32B	50.30	0.11	4.89	18.43
LLMs (few-shot)				
DeepSeek-R1	51.81	5.65	17.93	25.13
Gemini-2.5-flash (CoT)	64.95	10.87	61.42	45.75
Gemini-2.5-flash	74.05	39.90	62.97	58.97
Qwen2.5-72B-Instruct	47.17	16.85	14.61	26.21
Qwen3-32B	53.64	0.02	38.39	30.68
PLM (SFT)				
CoCoDet	98.94	97.41	98.37	98.24

Table 2: Performance on the ternary classification task. We report the per-class and average F1-scores (%).

Detector	Predicted AI Rate			Acc \uparrow	Sty-Rob
	Human \downarrow	Mix	AI \uparrow		
Radar	24.91	26.33	34.93	55.01	✓
LLMDet	98.82	98.45	99.26	50.22	✗
FastDetectGPT	53.09	92.98	92.56	69.74	✗
Binoculars(accuracy)	15.86	66.96	74.32	79.23	✓
Binoculars(low-fpr)	3.30	34.78	49.81	73.26	✓
LLM-DetectAIve	3.92	33.89	83.52	89.80	✓
CoCoDet	1.31	-	96.90	97.80	-

Table 3: Performance of general detectors. ACC is the mean accuracy for Human and AI sets (%). Sty-Rob indicates the style robustness.

hind CoCoDet (see Appendix E.4 for details). This confirms that the content-concentrated detection task cannot be solved simply by scaling up few-shot examples. An unexpected finding is the degraded performance of Gemini in its thinking mode compared to its non-thinking mode. To investigate the cause, we analyze the reasoning processes of Qwen3 and DeepSeek, as Gemini’s reasoning process is inaccessible. We find that despite being explicitly prompted to judge based on substantive content, the reasoning frequently defaults to analyzing textual style (e.g., overly polished transitions or formulaic phrasing). This tendency to fixate on stylistic cues, even when instructed otherwise, appears to be a key factor limiting the performance of LLMs on this task.

General detectors’ performance is dictated by style-robustness. Our benchmark against general detectors, presented in Table 3, reveals significant performance disparities and an overall inadequacy on this task. None of the general detectors achieve an average accuracy of 90%, and their classification of mix samples lacks consistency. To explain these

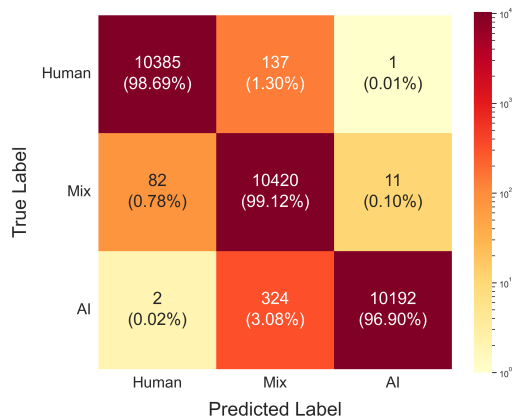


Figure 2: Confusion matrix of CoCoDet on the ternary classification task. The model exhibits high accuracy, with predictions clustered along the main diagonal. Errors are mostly confined to adjacent classes, while critical errors between the Human and AI classes are rare.

results, we examine the style-robustness of each detector. As defined in our experimental setup, this metric reflects a model’s ability to classify based on substantive content composition rather than superficial stylistic features. Methods explicitly designed to be style-robust, such as Radar, Binoculars, and LLM-DetectAIve, successfully meet our monotonicity criterion. We observe a correlation between this style-robustness and accuracy. The robust detectors, LLM-DetectAIve and Binoculars, outperform those non-robust like LLMdet and Fast-DetectGPT. A notable exception is Radar. While style-robust, it shows limited accuracy, which we attribute to its lack of training data from the review domain.

CoCoDet achieves a comprehensive state-of-the-art detection performance. CoCoDet achieves state-of-the-art results across all evaluated metrics, demonstrating its superiority over different classes of baseline models. Compared with LLM-based detectors, CoCoDet achieves a macro F1-score of 98.24% as seen in Table 2, drastically outperforming the best-performing few-shot LLM by nearly 40 percentage points. When compared to general detectors, it also secures a top accuracy of 97.80% as seen in Table 3, coupled with an exceptionally low false positive rate on human text of a mere 1.31%. Further analysis of the confusion matrix (Figure 2) reveals that critical errors between the *Human* and *AI* classes are virtually eliminated. It directly validates the effectiveness of our CSM Loss, which is specifically designed to penalize these severe errors.

Model Configuration	Human	Mix	AI	Average
CoCoDet (Full Model)	98.94	97.41	98.37	98.23
<i>Ablation of Auxiliary Tasks:</i>				
– Content Source	98.73	96.74	97.77	97.75
– Textual Style	98.72	96.74	97.86	97.77
– Collaboration Mode	98.73	95.77	96.44	96.98
<i>Ablation of Main Task:</i>				
– Base Margin	99.17	95.02	95.40	96.53
– Cost Margin	98.87	94.98	95.55	96.47
<i>Ablation of All Components:</i>				
– Margins & Aux.	98.43	94.30	95.18	95.97

Table 4: Ablation study of CoCoDet shows the benefit of each component of the framework in F1-score (%)

4.3 Generalization and Robustness

To validate the reliability of CoCoDet beyond the standard test set, we conduct extensive evaluations on out-of-distribution generalization and robustness. First, on 3,242 reviews generated by three unseen LLMs (Doubao, Kimi, and Mini-Max), CoCoDet achieves accuracies of 97.10%, 97.37%, and 85.04%, respectively. Second, in adversarial scenarios, the model correctly identifies **96.0%** of AI reviews manually humanized by experts, and achieves 100% accuracy on human reviews back-translated via two unseen intermediate languages. Finally, stratified analysis demonstrates consistent performance across short (98.91%), medium (98.62%), and long (97.19%) length bins, ruling out length-based heuristics. **Collectively, these results confirm CoCoDet’s robust detection accuracy across diverse unseen models and challenging perturbation scenarios.** Detailed results are provided in Appendix E.1.

4.4 Ablation Study

To validate the effectiveness and contribution of each component of CoCoDet, we conduct an ablation study. We systematically create several ablated versions of our model by removing key components one at a time: the auxiliary task losses and the core elements of our main loss function. The experimental setup for each ablated model is kept identical to that of the full CoCoDet model. Each ablated variant is fine-tuned for 5 epochs, and the model checkpoint that achieves the best performance on the validation set is selected for the final evaluation. The ablation results are detailed in Table 4.

All auxiliary tasks contribute to the final performance. The observation that removing either the content source attribution or textual style attri-

bution results in a modest performance drop is not a sign of redundancy. Instead, it highlights their complementary nature in creating an implicit disentanglement, where learning to isolate content inherently helps identify style, and vice versa. In contrast, the more substantial decline observed upon removing the collaboration mode attribution validates its orthogonal role.

Both margin factors in main task contribute to the detection effectiveness. Removing the Base Margin markedly impairs performance, indicating that a universal large decision boundary is fundamental for preventing classification ambiguity. The Cost Margin proves even more vital, as its removal results in the largest performance degradation across all experiments. The targeted penalty of the cost margin is essential for resolving the critical distinction between human and AI.

Finally, removing both margins and auxiliary tasks results in a performance drop, confirming their collective necessity.

4.5 AI Usage Trends in Post-ChatGPT Reviews

Since these anonymous reviews lack ground truth labels, our analysis relies on maintaining an extremely low false positive rate (FPR) on verified human text. As a baseline, we first analyze reviews from ICLR 2023 (pre-ChatGPT), where CoCoDet yields an FPR below 1% for *Any AI Involvement* and zero false positives for the *Mix* and *Pure AI Content* classes. To further validate this reliability, we also evaluate CoCoDet on the PeerRead dataset (Kang et al., 2018), which comprises purely human-written reviews from 2013 to 2017. Following the identical filtering pipeline used for CoCoNUTS, CoCoDet misclassified only 1 out of 8,624 valid human reviews, yielding an FPR of 0.011%. These consistent results provide strong evidence that CoCoDet is robust to temporal shifts and strictly avoids misclassifying human written reviews.

In contrast, reviews from the post-ChatGPT era show a clear increase in AI usage. The results indicate that AI assistance in peer review has become common, as reflected by the high proportion of *Any AI Involvement* across recent conferences. This usage is mainly limited to language polishing, as shown by the large gap between *Any AI Involvement* and the other classes. However, a more concerning trend also emerges: the share of *Pure AI Content* increases year by year, indicating that

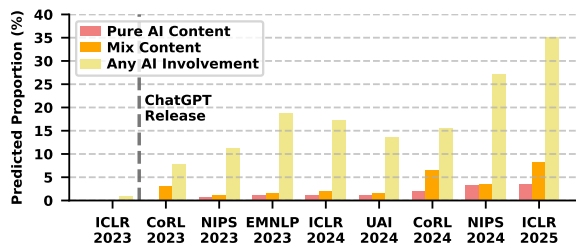


Figure 3: The predicted AI involvement of recent conference reviews. Pure AI Content and Mix Content correspond to the outputs of content composition identification. Any AI Involvement aggregates all categories except “HW” by the collaboration mode attribution.

fully AI-generated reviews are being submitted in practice. In addition, our manual analysis identifies another frequent issue: the paper summary section often shows signs of AI generation, and the risks of this practice warrant further attention.

5 Related Work

AI’s Role and Risks in Peer Review Recent research indicates a growing use of LLMs in peer review, extending beyond language polishing to the substantive modification of content (Zhou et al., 2025; Liang et al., 2024). While LLMs often serve as compensatory tools under deadline pressure, their application poses severe integrity risks. Studies consistently show that LLM-generated reviews frequently exhibit hallucinations, self-contradictions (Du et al., 2024), and a lack of technical soundness (Zhou et al., 2024a). In response, leading AI conferences have enacted policies permitting AI-assisted polishing but explicitly banning substantive generation. This urgently necessitates benchmarks and detectors capable of distinguishing permissible refinement from prohibited generation.

Benchmarks of AI-Generated Text Detection

Benchmarks for AI-generated text detection have progressed from simple binary classification (Guo et al., 2023) to more nuanced paradigms like source attribution (Dugan et al., 2023) and fine-grained classification (Wang et al., 2024a). However, these datasets largely overlook the deeply intertwined nature of modern human-AI collaborative writing in the unique context of academic peer review (Zhang et al., 2024). While a recent work targets this specific domain and highlights concerns about AI editing (Yu et al., 2025), its released dataset remains confined to a binary scenario, failing to capture the

complexity of real-world collaborative workflows.

AI-Generated Text Detectors General AI-generated text detectors are broadly divided into model-based and metric-based approaches. Model-based methods frame detection as a supervised classification task by fine-tuning pretrained language models (Abassy et al., 2024). Metric-based methods analyze properties such as conditional probability curvature (Bao et al., 2024) and perplexity (Hans et al., 2024). Recent works have begun adapting these for peer review, utilizing term frequency (Kumar et al., 2024), embeddings (Yu et al., 2025), or sentence-level LLM completion (Kumar et al., 2025). However, these approaches still rely heavily on fragile stylistic cues. Consequently, they fundamentally fail to differentiate compliant human authors who use AI for formatting from those who illegitimately generate substantive content.

6 Conclusion

In this work, we propose a content-based paradigm for AI review detection, introducing the CoCoNUTS benchmark and CoCoDet detector. Experiments validate the effectiveness of the paradigm. We find that baseline detectors are ill-suited for this task. In contrast, CoCoDet achieves outstanding performance with over 98% accuracy. Applying CoCoDet to recent reviews, we reveal widespread AI polishing and a concerning rise in fully machine-generated reviews. These observations underscore the urgent need for transparent mechanisms to identify AI involvement in scientific evaluation. The goal of this work is to provide a basis for attributing AI use in the peer review process and guide the responsible integration of AI into the scholarly ecosystem.

Limitations

Our work is subject to several limitations that we aim to address in subsequent studies. First, our model’s generalization is constrained by data scarcity. Its training data consists exclusively of English-language peer reviews from computer science, as the reviews from other domains are generally publicly unavailable. Furthermore, our dataset may not fully capture the sophistication of real-world generation scenarios. While our CoCoNUTS benchmark was designed to cover a wide variety of usage patterns, the sheer diversity of real-world AI application means some authentic workflows may not be represented. The model’s performance

against such methods that fall outside our simulations remains an important area for future validation.

Acknowledgements

We sincerely thank the reviewers for their insightful comments and valuable suggestions. This work was supported by the Natural Science Foundation of China (No. 62536008, 62572456, 62506354, 62272439), the Postdoctoral Fellowship Program of CPSF under Grant Number GZC20251041.

References

- Mervat Abassy, Kareem Elozeiri, Alexander Aziz, Minh Ngoc Ta, Raj Vardhan Tomar, Bimarsha Adhikari, Saad El Dine Ahmed, Yuxia Wang, Osama Mohammed Afzal, Zhuohan Xie, Jonibek Mansurov, Ekaterina Artemova, Vladislav Mikhailov, Rui Xing, Jiahui Geng, Hasan Iqbal, Zain Muhammad Mujahid, Tarek Mahmoud, Akim Tsvigun, and 5 others. 2024. **LLM-DetectAIve: a tool for fine-grained machine-generated text detection**. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 336–343, Miami, Florida, USA. Association for Computational Linguistics.
- ACL Rolling Review. ACL Rolling Review reviewer guidelines. <https://aclrollingreview.org/reviewerguidelines#q-can-i-use-generative-ai>. Accessed: 2025-06-01.
- Ekaterina Artemova, Jason S Lucas, Saranya Venkatraman, Jooyoung Lee, Sergei Tilga, Adaku Uchendu, and Vladislav Mikhailov. 2025. **Beemo: Benchmark of expert-edited machine-generated outputs**. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6992–7018, Albuquerque, New Mexico. Association for Computational Linguistics.
- Guangsheng Bao, Lihua Rong, Yanbin Zhao, Qiji Zhou, and Yue Zhang. 2025. **Decoupling content and expression: Two-dimensional detection of ai-generated text**. *Preprint*, arXiv:2503.00258.
- Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. **Fast-detectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature**. In *The Twelfth International Conference on Learning Representations*.
- Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2024. **Nougat: Neural optical understanding for academic documents**. In *The Twelfth International Conference on Learning Representations*.

- Lucio La Cava and Andrea Tagarelli. 2025. [Openturing-bench: An open-model-based benchmark and framework for machine-generated text detection and attribution](#). *Preprint*, arXiv:2504.11369.
- Tuhin Chakrabarty, Philippe Laban, and Chien-Sheng Wu. 2025. Can ai writing be salvaged? mitigating idiosyncrasies and improving human-ai alignment in the writing process through edits. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–33.
- Zihao Cheng, Li Zhou, Feng Jiang, Benyou Wang, and Haizhou Li. 2025. [Beyond binary: Towards fine-grained LLM-generated text detection via role recognition and involvement measurement](#). In *THE WEB CONFERENCE 2025*.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Jiangshu Du, Yibo Wang, Wenting Zhao, Zhongfen Deng, Shuaiqi Liu, Renze Lou, Henry Peng Zou, Pranav Narayanan Venkit, Nan Zhang, Mukund Srinath, Haoran Ranran Zhang, Vipul Gupta, Yinghui Li, Tao Li, Fei Wang, Qin Liu, Tianlin Liu, Pengzhi Gao, Congying Xia, and 21 others. 2024. [LLMs assist NLP researchers: Critique paper \(meta-\)reviewing](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5081–5099, Miami, Florida, USA. Association for Computational Linguistics.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.
- Liam Dugan, Alyssa Hwang, Filip Trhlfk, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. [RAID: A shared benchmark for robust evaluation of machine-generated text detectors](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12463–12492, Bangkok, Thailand. Association for Computational Linguistics.
- Liam Dugan, Daphne Ippolito, Arun Kirubarajan, Sherry Shi, and Chris Callison-Burch. 2023. [Real or fake text?: Investigating human ability to detect boundaries between human-written and machine-generated text](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):12763–12771.
- Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. [How close is chatgpt to human experts? comparison corpus, evaluation, and detection](#). *Preprint*, arXiv:2301.07597.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. [Spotting LLMs with binoculars: Zero-shot detection of machine-generated text](#). In *Forty-first International Conference on Machine Learning*.
- Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2024. [Mgtbench: Benchmarking machine-generated text detection](#). In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS '24*, page 2251–2265, New York, NY, USA. Association for Computing Machinery.
- Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. [RADAR: Robust AI-text detection via adversarial learning](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- ICLR. Policies on large language model usage at ICLR. <https://blog.iclr.cc/2025/08/26/policies-on-large-language-model-usage-at-iclr-2026/>. Accessed: 2025-06-01.
- ICML. ICML LLM policy. <https://icml.cc/Conferences/2026/Intro-LLM-Policy>. Accessed: 2025-06-01.
- Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. [A dataset of peer reviews \(PeerRead\): Collection, insights and NLP applications](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1647–1661, New Orleans, Louisiana. Association for Computational Linguistics.
- 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](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 27469–27500. Curran Associates, Inc.
- Sandeep Kumar, Samarth Garg, Sagnik Sengupta, Tirthankar Ghosal, and Asif Ekbal. 2025. [MixRevDetect: Towards detecting AI-generated content in hybrid peer reviews](#). In *Proceedings of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 944–953, Albuquerque, New Mexico. Association for Computational Linguistics.
- Sandeep Kumar, Mohit Sahu, Vardhan Gacche, Tirthankar Ghosal, and Asif Ekbal. 2024. ['quis](#)

- custodiet ipsos custodes?’ who will watch the watchmen? on detecting ai-generated peer-reviews. *Preprint*, arXiv:2410.09770.
- Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, Daniel McFarland, and James Y. Zou. 2024. **Monitoring AI-modified content at scale: A case study on the impact of chat-GPT on AI conference peer reviews**. In *Forty-first International Conference on Machine Learning*.
- Ilya Loshchilov and Frank Hutter. 2019. **Decoupled weight decay regularization**. In *International Conference on Learning Representations*.
- Samuel J Paech. 2025. Eq-bench creative writing benchmark v3. <https://github.com/EQ-bench/creative-writing-bench>.
- Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2025. **Can ai-generated text be reliably detected?** *Preprint*, arXiv:2303.11156.
- Shoumik Saha and Soheil Feizi. 2025. **Almost ai, almost human: The challenge of detecting ai-polished writing**. *Preprint*, arXiv:2502.15666.
- Zhenpeng Su, Xing Wu, Wei Zhou, Guangyuan Ma, and Songlin Hu. 2024. **Hc3 plus: A semantic-invariant human chatgpt comparison corpus**. *Preprint*, arXiv:2309.02731.
- Minh Ngoc Ta, Dong Cao Van, Duc-Anh Hoang, Minh Le-Anh, Truong Nguyen, My Anh Tran Nguyen, Yuxia Wang, Preslav Nakov, and Sang Dinh. 2025. **Faid: Fine-grained ai-generated text detection using multi-task auxiliary and multi-level contrastive learning**. *Preprint*, arXiv:2505.14271.
- Zhen Tao, Yanfang Chen, Dinghao Xi, Zhiyu Li, and Wei Xu. 2024. **Towards reliable detection of llm-generated texts: A comprehensive evaluation framework with cudrt**. *Preprint*, arXiv:2406.09056.
- Qwen Team. 2024. **Qwen2.5: A party of foundation models**.
- Qwen Team. 2025. **Qwen3 technical report**. *Preprint*, arXiv:2505.09388.
- 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**. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2001–2016, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. **Cosface: Large margin cosine loss for deep face recognition**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Pucetti, Thomas Arnold, Alham Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024a. **M4GT-bench: Evaluation benchmark for black-box machine-generated text detection**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3964–3992, Bangkok, Thailand. Association for Computational Linguistics.
- Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024b. **M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection**. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1369–1407, St. Julian’s, Malta. Association for Computational Linguistics.
- Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. **Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference**. *Preprint*, arXiv:2412.13663.
- 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 *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Rui Ye, Xianghe Pang, Jingyi Chai, Jiaao Chen, Zhenfei Yin, Zhen Xiang, Xiaowen Dong, Jing Shao, and Siheng Chen. 2024. **Are we there yet? revealing the risks of utilizing large language models in scholarly peer review**.
- Sungduk Yu, Man Luo, Avinash Madusu, Vasudev Lal, and Phillip Howard. 2025. **Is your paper being reviewed by an llm? benchmarking ai text detection in peer review**.
- Qihui Zhang, Chujie Gao, Dongping Chen, Yue Huang, Yixin Huang, Zhenyang Sun, Shilin Zhang, Weiye Li, Zhengyan Fu, Yao Wan, and Lichao Sun. 2024. **LLM-as-a-coauthor: Can mixed human-written and machine-generated text be detected?** In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 409–436, Mexico City, Mexico. Association for Computational Linguistics.
- Li Zhou, Ruijie Zhang, Xunlian Dai, Daniel Herscovich, and Haizhou Li. 2025. **Large language models penetration in scholarly writing and peer review**. *Preprint*, arXiv:2502.11193.

Ruiyang Zhou, Lu Chen, and Kai Yu. 2024a. [Is LLM a reliable reviewer? a comprehensive evaluation of LLM on automatic paper reviewing tasks](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9340–9351, Torino, Italia. ELRA and ICCL.

Ying Zhou, Ben He, and Le Sun. 2024b. [Humanizing machine-generated content: Evading AI-text detection through adversarial attack](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8427–8437, Torino, Italia. ELRA and ICCL.

A Ethical Considerations

Our work is motivated by the ethical imperative to uphold academic integrity by distinguishing substantive AI generation from permissible language polishing. However, we acknowledge the inherent risks of false positives and the potential for dual-use. To ensure responsible deployment, we emphasize that CoCoDet is intended primarily for aggregate monitoring or as an assistive screening tool, rather than for automated enforcement. It must never serve as a final arbiter.

B Content and Style

The conceptual distinction between content and style is foundational to our work. We posit that reliable AI-generated review detection should prioritize content over style. This premise necessitates clear definitions for these two concepts and a practical strategy to operationalize them in our model training.

B.1 Definition of Content

We define content as the invariant semantic core of a text, that is, the underlying meaning, logical propositions, factual claims, and expressed ideas. Content addresses the question, “What is said?” It represents the informational payload that should, in principle, remain invariant across different phrasings or translations. In the context of peer review, this includes the reviewer’s critical assessments, their summary of a paper’s contributions, and specific, actionable recommendations.

Our central hypothesis is that while style is highly malleable, the provenance of complex, domain-specific ideas serves as a more reliable signal of authorship. Prior studies have demonstrated that while LLMs can generate superficial critiques, they often differ significantly from human experts in the depth of specific feedback and critical reasoning (Du et al., 2024; Zhou et al., 2024a). Therefore, our detection task is framed as a problem of content composition identification.

B.2 Definition of Style

Conversely, we define style as the set of formal properties governing how content is expressed. Style answers the question, “How is it being said?” and encompasses a wide spectrum of features that can be altered without fundamentally altering the core meaning. We categorize these features into

a three-level hierarchy to serve as a conceptual framework:

- **Linguistic Features** This level constitutes the foundational lexical choices (e.g., vocabulary, formality, synonym preference) and syntactic patterns (e.g., sentence length, complexity, use of active vs. passive voice).
- **Discourse-Level Features** This higher level relates to the text’s overall structure and rhetorical strategy. It includes the ordering of arguments, the use of logical transitions, and the persuasive or critical tone of the writing.
- **Statistical Artifacts** This category covers the subtle and subconscious statistical regularities that act as a statistical signature of the author for both human or a specific LLM. These include low-level patterns like perplexity (PPL) and n-gram frequency distributions, as well as more overt indicators like characteristic boilerplate phrases.

B.3 Implementation of Content and Style in CoCoDet Training

While the definitions above provide the theoretical grounding, manually engineering features or creating granular labels for these concepts is computationally intractable for large-scale training. Therefore, we implement these concepts in CoCoDet using a data-driven Proxy Strategy:

Implementation of Content To capture content, we employ Content Source Attribution as a proxy. This strategy relies on the premise that distinct authoring entities possess unique cognitive features regarding what they write. These entities include both human experts and specific AI models, such as Qwen2.5-72B and Qwen3-32B. Such fingerprints manifest as observable differences in knowledge boundaries, reasoning capabilities, and the depth of critique. By training the model to attribute text to the specific authoring entity, we compel it to learn the underlying semantic distinctions that constitute content.

Implementation of Style To capture style, we utilize Textual Style Attribution as a proxy. This approach is guided by empirical findings that models from the same family, such as the Llama family or the Qwen family, exhibit consistent stylistic

signatures across the hierarchy described in Appendix B.2. Specifically, these models share similar vocabulary distributions and syntactic preferences because they rely on shared architectures and training data (Ta et al., 2025; Paech, 2025). By training the model to attribute text to a generic model family, we effectively enable the detector to abstract away from specific content and focus on the shared linguistic and statistical patterns that constitute style.

C Dataset Details

C.1 Prompt for Dataset Construction

We use the following prompts for review generation:

HW&MT

EN2CN:

You are a professional AI field translator. Please translate the following English peer review into Chinese, paying attention to:

1. Technical terms should be accurately translated
2. Maintain the original review structure
3. Keep the academic rigor while making it fluent in Chinese
4. Output only the translated text, without any other information

Review: {en_review}

CN2EN:

You are a professional AI field translator. Please translate the following Chinese peer review into English, paying attention to:

1. Technical terms should be accurately translated
2. Maintain the original review structure
3. Keep the academic rigor while making it fluent in English
4. Output only the translated text, without any other information

Review: {cn_review}

HW&MP

You are a senior AI researcher and experienced reviewer for top-tier AI conferences. Please polish the following peer review. Please maintain the original technical content and core evaluation while improving sentence structure, terminology consistency,

and readability.

Review: {review}

Only output the polished review, do not include any other details.

HW&MG

You are a senior AI researcher and an experienced reviewer for top-tier AI conferences. Your task is to polish and expand a user-provided peer review. Your goal is to elevate it into a high-quality, professional piece by:

1. Delete some redundant content to make it more concise.
2. Expanding its content based on the provided paper content.
3. Improve the sentence structure, terminology consistency, and readability.
4. Output: Provide only the raw text of the elevated review, do not include any other details.

Review: {review}

Paper content : {paper_content}

MG

You are a senior AI researcher and experienced reviewer for top-tier AI conferences. Please carefully read the example reviews and then analyze the paper content provided by the user. After that write a comprehensive and objective review of the paper. Please follow the basic review content requirements(e.g., summary, evaluation, questions, suggestions for improvement) and ground your evaluation in the provided paper content.

Here are two examples of reviews:

Example 1: {example1}

Example 2: {example2}

Paper content : {paper_content}

Please only output the review, do not include any other details.

MG&MP

You are a senior AI researcher and experienced reviewer for top-tier AI conferences. Please paraphrase the review given by the

user to make it more natural and human-written.

Review: {review}

Only output the paraphrased review, do not include any other details.

C.2 Dataset Composition and Partition

Dataset overview First, we present the overall composition of the CoCoNUTS dataset in Table 5. The dataset is categorized into three classes (Human, Mix, AI) and further broken down by fine-grained collaboration modes.

Human				105,180
HW		HWMT		
Human	Llama	Qwen2.5		
35,060	17,142	17,918		
Gemini		Llama	Qwen3	
10,372		12,450	12,238	
Mix				105,180
Gemini		Llama	Qwen2.5	Qwen3
19,251		43,997	29,201	12,731
AI				105,175
Claude*		DeepSeek	Gemini	GPT*
7,578		3,500	8,000	7,597
Llama		Qwen2.5	Qwen3	
13,500		10,000	10,000	
MGMP				
Gemini-Llama	Llama-Gemini	Llama-Qwen2.5	DS-Gemini	
7,000	8,000	7,000	3,000	
DS-Llama	Qwen2.5-Gemini	Qwen3-Gemini	Qwen3-Llama	
5,000	4,000	5,000	6,000	

Table 5: Dataset statistics. * Data sourced from the AI-Peer-Review-Detection-Benchmark.

Dataset split We partition the complete CoCoNUTS dataset into training, validation, and test sets using an approximate 8:1:1 ratio. To ensure that each set is a representative sample of the overall data distribution, we employ a stratified sampling strategy. The stratification is performed based on the generating or modifying LLM for each instance. This approach guarantees that the data from every language model used in our construction process is proportionally represented across the training, validation, and test sets.

This model-level stratification is crucial for preventing a model from being evaluated on a significantly different distribution of AI-generated styles than it was trained on. It ensures that our evaluation robustly measures the detector’s ability to handle

a consistent and diverse mix of AI sources. The detailed statistics for each split are presented in Table 6, Table 7, and Table 8.

Human				84,104
HW		HWMT		
Human	Llama	Qwen2.5		
28,069	13,724	14,313		
		HWMP		
Gemini	Llama	Qwen3		
8,287	9,938	9,773		
Mix				84,184
		HWMG		
Gemini	Llama	Qwen2.5	Qwen3	
15,392	35,230	23,393	10,169	
AI				84,139
		MG		
Claude	DeepSeek	Gemini	GPT4o	
6,062	2,800	6,400	6,077	
Llama	Qwen2.5	Qwen3		
10,800	8,000	8,000		
		MGMP		
Gemini-Llama	Llama-Gemini	Llama-Qwen2.5	DS-Gemini	
5,600	6,400	5,600	2,400	
DS-Llama	Qwen2.5-Gemini	Qwen3-Gemini	Qwen3-Llama	
4,000	3,200	4,000	4,800	

Table 6: Statistics of our training set.

Human				10,553
HW		HWMT		
Human	Llama	Qwen2.5		
3,495	1,703	1,815		
		HWMP		
Gemini	Llama	Qwen3		
1,050	1,264	1,226		
Mix				10,483
		HWMG		
Gemini	Llama	Qwen2.5	Qwen3	
1,923	4,379	2,905	1,276	
AI				10,518
		MG		
Claude	DeepSeek	Gemini	GPT4o	
758	350	800	760	
Llama	Qwen2.5	Qwen3		
1,350	1,000	1,000		
		MGMP		
Gemini-Llama	Llama-Gemini	Llama-Qwen2.5	DS-Gemini	
700	800	700	300	
DS-Llama	Qwen2.5-Gemini	Qwen3-Gemini	Qwen3-Llama	
500	400	500	600	

Table 7: Statistics of our validation set.

C.3 Data Scope and Purity Guarantee

To ensure a rigorous and uncontaminated baseline, all human-written reviews utilized for constructing the CoCoNUTS benchmark, and consequently used for training and evaluating CoCoDet, are derived exclusively from pre-ChatGPT submissions

Human				10,523
HW		HWMT		
Human	Llama	Qwen2.5		
3,496	1,715	1,790		
		HWMP		
Gemini	Llama	Qwen3		
1,035	1,248	1,239		
Mix				10,513
		HWMG		
Gemini	Llama	Qwen2.5	Qwen3	
1,936	4,388	2,903	1,286	
AI				10,518
		MG		
Claude	DeepSeek	Gemini	GPT4o	
758	350	800	760	
Llama	Qwen2.5	Qwen3		
1,350	1,000	1,000		
		MGMP		
Gemini-Llama	Llama-Gemini	Llama-Qwen2.5	DS-Gemini	
700	800	700	300	
DS-Llama	Qwen2.5-Gemini	Qwen3-Gemini	Qwen3-Llama	
500	400	500	600	

Table 8: Statistics of our test set.

(i.e., in or before 2022). This strict temporal cut-off guarantees that our benchmark is built upon a purely human-authored foundation, effectively eliminating the risk of undetected AI-generated texts masquerading as human writing.

It is important to note the specific handling of ICLR 2023 reviews. Although associated with the 2023 conference cycle, the official review submission deadline for ICLR 2023 preceded the public release of ChatGPT. We deliberately excluded these reviews from the CoCoNUTS benchmark (i.e., they are not used for model training or testing). Instead, we specifically held them out as an out-of-distribution (OOD), verified pure-human baseline to independently validate the reliability and false-positive rate of CoCoDet (as detailed in Section 4.5).

Furthermore, reviews submitted in the actual post-ChatGPT era (later 2023 to 2025) are strictly reserved for the real-world trend analysis discussed in Section 4.5. These recent reviews are completely excluded from dataset construction, model training, and benchmark evaluation processes to prevent any potential label contamination.

C.4 Review Length Statistics

To provide a comprehensive statistical overview of our CoCoNUTS dataset, we present a detailed analysis of its textual properties, focusing on length and distribution. These statistics underscore the distinct characteristics of the six fine-grained modes

and the three content-based classes, which are central to our benchmark’s design. This analysis reveals key differences in text length and structure across categories, which may influence detector performance and highlight the challenges of distinguishing between different forms of human and AI collaboration.

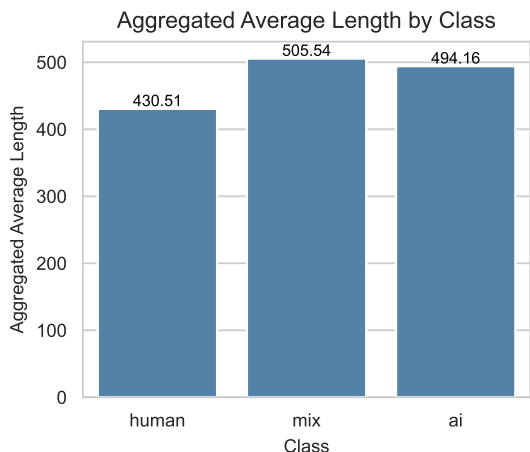


Figure 4: Aggregated average length (in words) across the three content-composition classes: Human, Mix, and AI. Peer reviews with substantive content generated by AI demonstrate longer length.

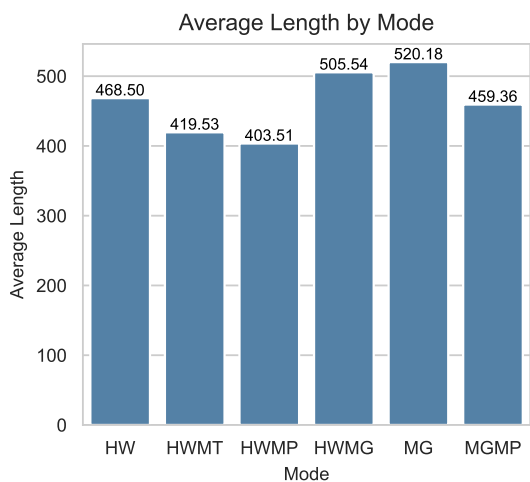


Figure 5: Average text length (in words) for each of the six collaboration modes. Modes involving direct machine generation (MG, HWMG) tend to be longer, while post-processing steps like polishing (HWMP) or paraphrasing (MGMP) can result in shorter texts compared to their respective source modes.

We present a statistical analysis of the average text length across different classes in our CoCoNUTS dataset, revealing systematic variations based on the nature of AI involvement. As illus-

trated in Figure 4, reviews with substantive AI contributions exhibit a greater length than purely human-authored texts. Specifically, the Human class has an average length of approximately 431 words, whereas the Mix and AI classes are substantially longer, at 506 and 494 words, respectively. This suggests that the process of content generation by large language models, whether partial or complete, tends to produce more verbose outputs compared to the human baseline.

A more granular analysis, presented in Figure 5, clarifies the nuanced effects of different human-AI collaboration modes. The purely machine-generated mode produces the longest texts, averaging 520 words. However, a critical observation is that the semantic-invariant operations reduce text length. For instance, when human-written reviews (469 words) are subjected to machine polishing or machine translation, their average lengths decrease to 404 and 420 words, respectively. A similar condensing effect is observed when machine-generated texts are paraphrased, reducing the average length from 520 to 459 words.

D Experimental Setup Details

D.1 Training Hyper-parameters

The final hyper-parameters used to fine-tune our CoCoDet model, determined through a sequential grid search on the validation set, are detailed in Table 9. We successively tune: (1) the learning rate over $\{1e-5, 2e-5, \dots, 5e-5\}$; (2) the base and cost margins m_{base}, m_{cost} from $\{0.2, 0.25, 0.3\}$; and (3) the auxiliary task weights, by first jointly searching α and β over $\{0.2, 0.3, 0.4, 0.5\}$ and then tuning γ over $\{0.1, 0.2, 0.3, 0.4\}$. All experiments use a fixed random seed of 42 for reproducibility.

Parameter	Value
Backbone Model	ModernBERT-base
Epochs	5
Max Sequence Length	2048
Learning Rate	2e-5
Batch Size	16
Weight Decay	0.01
Auxiliary Weights (α, β, γ)	0.4, 0.4, 0.2
Base Margin	0.25
Cost Margin	0.25
Scaling Factor	30

Table 9: Final hyperparameters for fine-tuning the CoCoDet model. These values are selected based on the best performance on the validation set after a sequential grid search.

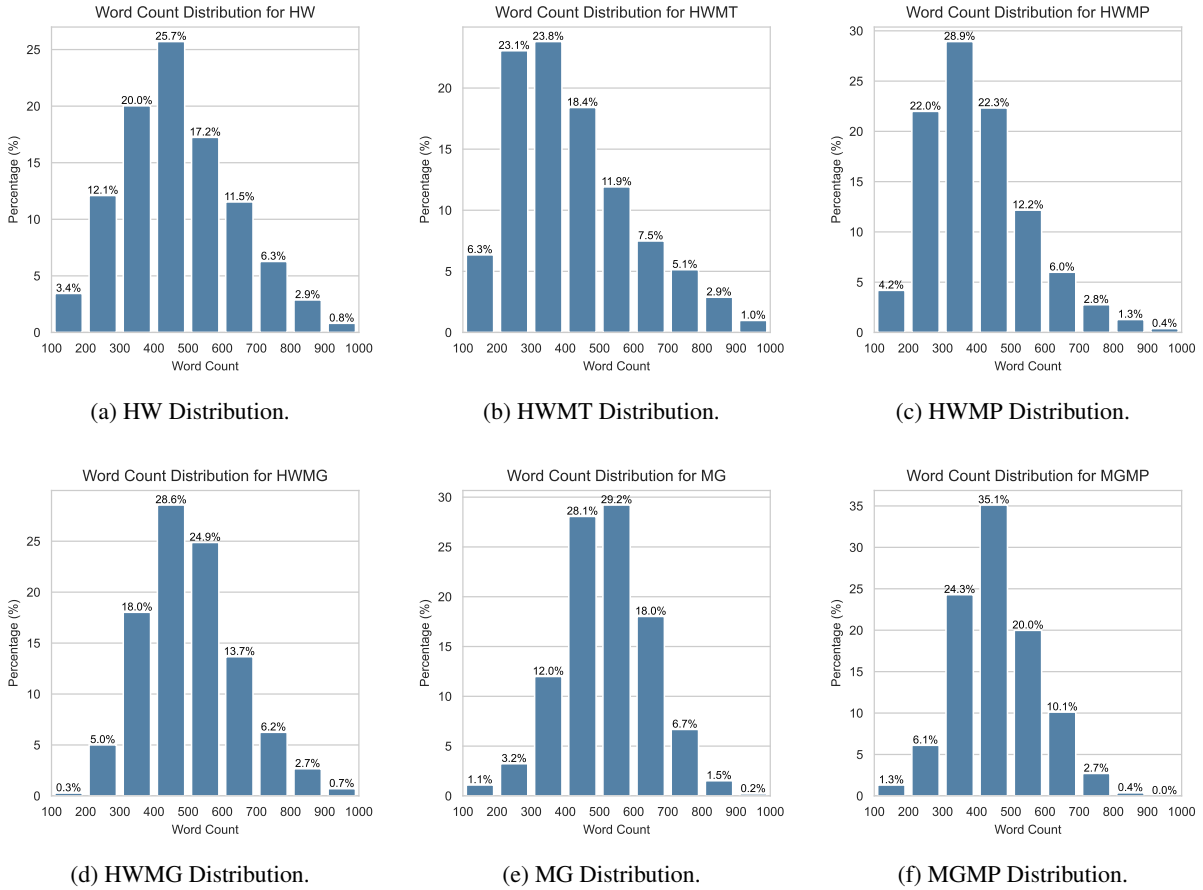


Figure 6: Word count distributions for each of the six fine-grained collaboration modes in the CoCoNUTS dataset. This detailed view reveals the distinct length characteristics of each mode, such as the longer tails for machine-generated content (e.g., MG, HWMG) and the condensing effect of polishing (HWMP).

D.2 Details of General Detectors

LLMDet (Wu et al., 2023): LLMDet provides a method for model-specific detection of AI-generated text without requiring real-time access to the source LLMs during inference. The approach operates in two phases. First, an offline ‘dictionary construction’ phase builds a fingerprint for each target LLM. This involves pre-computing and storing frequent n-gram patterns and their corresponding next-token probability distributions. During the online detection phase, LLMDet calculates a ‘proxy perplexity’ score for the input text against each pre-computed dictionary. These proxy scores are then used as features for a classifier to identify the most likely source model.

RADAR (Hu et al., 2023): RADAR is designed to address a critical vulnerability in AI text detectors: their susceptibility to paraphrasing attacks. It employs an adversarial training framework where a detector model and a paraphraser model are trained in opposition. The paraphraser’s objective is to

rewrite AI-generated text to evade detection, while the detector is simultaneously trained to correctly identify not only original AI text but also these adversarially paraphrased versions. This process forces the detector to learn more robust features that are resilient to stylistic modifications, thereby improving its performance in realistic, adversarial scenarios.

Fast-DetectGPT (Bao et al., 2024): Fast-DetectGPT presents a zero-shot detection method that significantly improves both the efficiency and accuracy of identifying machine-generated text. The core of the method is a novel metric called ‘conditional probability curvature.’ This metric is based on the hypothesis that machine-generated text exhibits a different statistical pattern in its conditional probability distribution compared to human-written text. Instead of re-evaluating multiple perturbed versions of a passage, Fast-DetectGPT computes this curvature by analyzing alternative token probabilities obtained from a sin-

gle forward pass of a scoring model, drastically reducing computational overhead.

Binoculars (Hans et al., 2024): Binoculars introduces a zero-shot detection method that identifies AI-generated text by analyzing it from two perspectives using two closely related language models: an “observer” and a “performer”. The core idea is to compute a ratio between the text’s standard perplexity (as seen by the observer) and its “cross-perplexity”, which measures the predictive divergence between the two models. This ratio creates a robust statistical signature that effectively distinguishes machine-generated text, which shows high model-to-model agreement, from human-written text, which exhibits greater variability. The method is notably effective at normalizing for unusual prompts that can otherwise mislead simpler perplexity-based detectors.

LLM-DetectAIve (Abassy et al., 2024): LLM-DetectAIve advances beyond binary classification by providing a fine-grained detection system. Instead of simply labeling text as human or machine, it distinguishes between four categories: purely human-written, purely machine-generated, machine-generated text that has been “humanized”, and human-written text that has been “polished” by a machine. This nuanced approach is designed to differentiate between acceptable uses of LLMs (e.g., polishing) and attempts to obfuscate AI authorship, making it particularly relevant for educational and academic contexts.

Detailed Evaluation Protocols for General Detectors To ensure a fair and rigorous comparison between our ternary formulation and existing general purpose detectors, we adopted specific operating protocols and label mapping rules during our evaluation:

Threshold Selection We strictly followed the officially recommended configurations for all metric-based baselines. For instance, Binoculars was evaluated at both its *High Accuracy* (optimized for F1) and *Low False Positive Rate* (FPR < 0.01%) thresholds, as these are the two operating points recommended by the authors. If no specific operating point was recommended, we utilized the default argmax prediction.

Label Mapping To align the outputs of fine-grained baselines with our ternary classes, we applied specific mapping rules. As LLMDet provides

probabilities for specific source models as well as a *Human_write* category, we mapped the *Human_write* prediction to our Human class, while all other specific AI source probabilities were aggregated into the AI class. Additionally, to facilitate a fair comparison for LLM-DetectAIve, which is a fine grained multi class model, we mapped its *HW* and *HWMP* predictions to the *Human* class, and its *MG* and *MGMP* predictions to the *AI* class.

D.3 Prompts for LLM-based Detectors

The exact prompts for LLM-based Detectors of both zero-shot and few-shot settings are shown below:

Zero-shot

You are an expert AI generated peer review detector. Your task is to classify the given text into one of three categories based on the content: Follow these rules precisely:

1. **‘human’**: Classify as ‘human’ if the core content was written by a human. This includes texts that were later machine-translated or polished by language tools.
2. **‘ai’**: Classify as ‘ai’ if the core content was generated by an AI. This includes texts that were later edited or ‘humanized’ by a person to sound more natural.
3. **‘mix’**: Classify as ‘mix’ only if the text contains substantive content contributions from both human and AI. This includes some sections written by a human and others generated by an AI.

Your response must be *only* one of these three words. Do not provide any explanations or additional text.

Few-shot

You are an expert AI generated peer review detector. Your task is to classify the given text into one of three categories based on the content: Follow these rules precisely:

1. **‘human’**: Classify as ‘human’ if the core content was written by a human. This includes texts that were later machine-translated or polished by language tools.
2. **‘ai’**: Classify as ‘ai’ if the core content was generated by an AI. This includes texts that were later edited or ‘humanized’ by a person to sound more natural.

3. **‘mix’**: Classify as ‘mix’ only if the text contains substantive content contributions from both human and AI. This includes some sections written by a human and others generated by an AI.

Your response must be *only* one of these three words. Do not provide any explanations or additional text.

Example for ‘human’:

“human_example”

Correct Answer: human

Example for ‘mix’:

“mix_example”

Correct Answer: mix

Example for ‘ai’:

“ai_example”

Correct Answer: ai

E Supplementary Experiments and Analysis

E.1 Detailed Generalization and Robustness Experiments

Generalization to Unseen Models: We evaluate CoCoDet on reviews generated by three unseen models: Doubao-Seed-1.6, Kimi-K2-Instruct-0905, and MiniMax-M2. To ensure the data quality, we apply the identical data processing pipeline used for the CoConuts dataset construction to all these external datasets. This pipeline includes the same quality filtering steps and length constraints detailed in Section 2.1.

The evaluation is conducted in two stages. First, we perform an initial pilot study using Doubao-Seed-1.6 with 100 raw generated samples. After passing through the standard processing pipeline, 68 valid reviews remained for evaluation. To provide a more rigorous assessment based on this preliminary success, we subsequently expand the scale by generating 2,500 raw reviews each from Kimi-K2-Instruct-0905 and MiniMax-M2. Applying the same filtering criteria results in 1,483 and 1,691 valid samples, respectively. As shown in Table 10, the model achieves consistent performance across these unseen generators regardless of the sample size.

Robustness to Independent Manual Rewrites: For adversarial evasion, we conducted a comprehensive evaluation using a total of 250 manually rewritten reviews. This dataset combines 50 samples manually paraphrased by the authors with an

Unseen Model	# Valid Reviews	Accuracy (%)
Doubao-Seed-1.6	68	97.10
Kimi-K2-Instruct-0905	1483	97.37
MiniMax-M2	1691	85.04
Overall (Weighted)	3242	90.93

Table 10: Generalization performance on unseen LLMs. All datasets underwent the same standard filtering pipeline used in CoConuts construction.

additional 200 samples rewritten by external experts. The external group consisted of three post-graduate students majoring in AI, all of whom are experienced with the submission, peer-review, and rebuttal process of major AI conferences. Among the external experts, one annotator edited 100 reviews, while the other two edited 50 each. Together with the authors, all annotators followed the same instruction to paraphrase the reviews by “replacing common AI vocabulary and altering AI writing structures, while ensuring the core content remained unchanged.” On this combined dataset of 250 challenged samples, CoCoDet successfully identifies 238 reviews as AI-generated, achieving an overall accuracy of **95.2%**. This demonstrates consistent resilience to human paraphrasing.

Generalization to HWMT via OOD intermediate languages: To ensure our model does not exhibit bias or unfairness against reviews that might originate from or be influenced by other languages, we conducted a back-translation experiment. We sample 1,000 human-written (HW) reviews and used Spanish and French—two languages not utilized in our training data construction—as intermediate languages. This process generated 2,000 samples to simulate scenarios where potential linguistic artifacts from translation or non-native writing might be introduced. CoCoDet demonstrated exceptional robustness by correctly classifying all 2,000 samples as human, achieving a perfect accuracy of **100.0%**. This result confirms that the model focuses on the underlying semantic content and remains robust against potential perturbations introduced by OOD intermediate languages, ensuring fairness across diverse linguistic contexts.

Robustness across Text Lengths: As detailed in Section 2.1, our dataset construction process automatically removes samples falling outside the 5th-95th percentile length of the HW set to exclude uninformatively short or excessively long reviews. To verify that our model does not rely on length heuristics within this valid range, we partitioned

the test set into three equal-sized bins based on word count: Short, Medium, and Long. The results in Table 11 demonstrate that CoCoDet maintains high accuracy across all length bins. The negligible performance variation confirms that the model relies on semantic signals rather than length.

Length Range	Word Count	Samples	Accuracy	Macro F1
Short	154 - 388	10,527	98.91%	0.9904
Medium	389 - 513	10,506	98.62%	0.9863
Long	514 - 944	10,521	97.19%	0.9731
Overall	154 - 944	31,554	98.24%	0.9833

Table 11: Performance stratified by text length.

E.2 Validation of Content-Based Detection on HWMP Reviews

Recent academic policies permit AI-assisted polishing but prohibit substantive generation. Addressing this boundary requires ensuring that classification relies on content composition rather than superficial stylistic features. Our CoCoNUTS benchmark is fundamentally designed to assess this capability, with the **Human-Written Machine-Polished (HWMP)** subset representing the most confounding and critical challenge for current detectors.

This subset serves as a prime example of the unresolved boundary in detection: it contains compliant, human-derived content wrapped in AI-modified style. Standard detectors, which heavily rely on stylistic heuristics, find this category particularly deceptive, often misclassifying such permissible refinement as illicit generation. This widespread failure leads to a high risk of false accusations against compliant authors.

To quantify this risk, we present a comparative analysis of the Predicted AI Rate (False Positive Rate) on the HWMP mode. High values here indicate a detector’s tendency to falsely accuse compliant authors.

Method	Predicted AI Rate
LLMDet	99.77%
FastDetectGPT	72.00%
Binoculars (ACC)	32.48%
Binoculars (low-FPR)	7.84%
LLM-DetectAIve	9.82%
CoCoDet (Ours)	
<i>Strict Protocol (AI + Mix)</i>	3.89%
<i>Standard Protocol (AI Only)</i>	0.03%

Table 12: Predicted AI Rate (False Positive Rate) on the HWMP category. This subset consists of permissible polished reviews.

As shown in Table 12, general detectors exhibit severe limitations in this challenging setting. Methods like LLMDet (99.77%) and FastDetectGPT (72.00%) misclassify the majority of polished texts. Even the robust baseline Binoculars (low-FPR) maintains a 7.84% error rate. In contrast, CoCoDet effectively disentangles content from style. Under the strict protocol (treating Mix predictions as errors), it maintains a low FPR of 3.89%. Notably, when considering only critical errors (classifying human content directly as AI), the error rate drops to 0.03% (1/3,522 samples). This validates that CoCoDet does not merely memorize style, but effectively disentangles content from style to achieve robust detection.

E.3 Quantitative and Qualitative Analysis of Content Divergence

Quantitative Analysis: To directly address the concern regarding the differentiability of human and AI content, we conducted a semantic similarity analysis. We sampled 100 pairs of human-written and AI-generated reviews for the same papers. To capture semantic meaning beyond simple lexical overlap, we employed Qwen3-Embedding-0.6B to represent each review as a vector and calculate their cosine similarity.

The average cosine similarity is found to be **0.7489**. This score is particularly revealing when considering the factors that should drive similarity higher: both reviews discuss the exact same paper, often include direct quotes from the source, and typically feature a highly similar summary section.

To provide a firm baseline for what “high similarity” looks like under these conditions, we conducted a control experiment measuring the similarity between reviews generated by two different LLMs (Gemini-2.5-flash and Qwen2.5-72B-Instruct) for 100 papers. The average similarity between these two AI models was **0.9025**.

This comparison clearly indicates that even after accounting for the shared topic and structural elements, the core content, reasoning paths, and critical insights in human vs. AI reviews are significantly more divergent than those between two distinct AI models. This provides strong quantitative evidence that AI models tend to converge on similar content patterns, whereas human reviewers exhibit unique, divergent perspectives.

Qualitative Case Study: To make these abstract differences concrete, we provide a case study comparing a review generated by **Gemini 3 Pro** against

three human-written reviews for the same paper.

As shown in the examples below, the AI-generated review is highly competent, structured, and provides valid critiques. However, it lacks the distinct signatures of human expertise and personal perspective. In contrast, the human reviewers display unique characteristics:

- **Human Review 1** states the NLP domain expertise to contextualize their critique.
- **Human Review 2** expresses personal inspiration as an “advocate of the sub-field”.
- **Human Review 3** focuses on minute details like an unreferenced figure.

These elements demonstrate that human reviews stem from diverse backgrounds and individual interests, creating the unique content signatures our CoCoDet model captures.

AI-Generated Review (Gemini 3 Pro)

This paper...However, several limitations are worth noting. Firstly, the method is fundamentally constrained by its reliance on a small set of top- k exemplars ($k = 15$). This approach assumes that a neuron’s function is entirely characterized by its highest activations, potentially missing subtler behaviors, negative inhibitions, or complex polysemanticity where the tail of the distribution matters. Secondly, while the method generalizes well to held-out architectures, it is still a supervised approach dependent on the quality and scope of the human annotations. If a neural network learns a feature that is fundamentally “alien” to human perception or outside the training distribution of Milannotations, the model may generate plausible-sounding but incorrect hallucinations. Finally, the use of LSTM-based architectures for the language model and decoder, while functional, is somewhat dated; modern Transformer-based architectures might yield better compositional performance and handle longer-range dependencies in complex descriptions more effectively.

Questions: 1. In the “Editing Spurious Features” experiment, the paper notes that 300 neurons were identified as text-related, but ablating only 13 resulted in the reported performance gain. How exactly were these spe-

cific 13 neurons selected from the larger set of 300? Was the importance score threshold determined heuristically, or was there an automated selection process? 2. How does MILAN handle neurons that are highly polysemantic (activating for two or more visually unrelated concepts)? While the paper discusses “max word difference,” does the generation model consistently produce conjunctive descriptions (e.g., “Concept A and Concept B”), or does it tend to mode-collapse onto a single concept or a vague generalization? 3. How sensitive is the quality of the generated descriptions to the number of exemplars (k)? Is $k = 15$ an empirical sweet spot, and does increasing this number provide diminishing returns or change the semantic content of the descriptions?

Suggestions for Improvement: 1. It would be beneficial to conduct an ablation study regarding the number of exemplars used. Understanding the trade-off between the number of images processed and the specificity of the resulting description would be valuable for practitioners balancing computational cost and accuracy. 2. The paper would benefit from a comparison against zero-shot labeling baselines using large pre-trained Vision-Language Models (e.g., using CLIP to rank a set of candidate descriptions or using a multimodal LLM). This would help contextualize the performance of the supervised MILAN approach against unsupervised foundation models. 3. A deeper qualitative analysis of failure cases would be insightful. Specifically, are there categories of visual features (e.g., high-frequency textures, abstract geometric patterns) that are systematically difficult for the model to describe compared to object-centric features?""

Human Review 1 (Domain Expert Perspective)

This work proposes a framework for interpreting model behaviour by generating language descriptions of neurons in the model. The method is I want to preface my review by stating that I am a domain expert in NLP and not CV. While I attempted a prelim-

inary search for related work, I am not sure if there is prior work in specifically generating language descriptions of neurons (I understand that there is prior work in explaining neuron explaining neuron behaviour by examining inputs that activate it). Hence I may significantly modify my review based on the response of my peer reviewers. Why is there so much variance across architecture pairs? For example large gains on AlexNet -> Places and ResNet -> ImageNet but small gains on ResNet -> Places and AlexNet -> ImageNet.

Human Review 2 (Personal Advocacy)

The authors introduced MILAN, for mutual-information-guided linguistic annotation of neurons) that automatically labels neurons with open-ended, compositional, natural language descriptions It is good read and inspirational to me since I have been an advocate of the sub-field. To be able go deeper on making use of the intermediate stages of a network, for visibility as well as new AI product features. The summary above shows the short of the discoveries. I have to experiment with the technique myself to go further. However, it looks promising. I also appreciate the amount of effort put on testing the system, on many architectures, datasets, and tasks. If there is more, I would like to have better characterization of the limits of the approach. Grammars, typos, etc.: pg.1, par.1: convlution —> convolution

Human Review 3 (Attention to Detail)

This paper describes a novel procedure (MILAN) to interpret deep learning models for computer vision by generating natural language description that specifies the activation selectivity of a given neuron in the model - Fig. 3 is not referenced in the main text. And I think these failure modes are interesting, and taking a closer look at them might be inspiring for improving this model in future studies. Do authors have further comments or thoughts about this result non-robust units and edit the network to im-

prove its performance? - Minor: - What do different dots refer to in Fig. 5?

E.4 Impact of Increasing Few-Shot Examples

To investigate whether providing more in-context examples could unlock the potential of LLMs, we extend our evaluation to include 4-shot, 6-shot, and 8-shot settings on representative models. The results are detailed in Table 13.

While increasing the number of shots yields marginal improvements for some models (e.g., DeepSeek-R1 and Llama-3.3), it fails to bring a fundamental qualitative leap. For instance, Qwen2.5 exhibits performance degradation and unstable class bias with more examples. Ultimately, even with extensive in-context guidance (8-shot), the best-performing LLM still lags behind CoCoDet.

Model	Shots	Human	Mix	AI	Average
DeepSeek-R1	4-shot	51.93	11.34	14.44	25.90
	6-shot	54.89	26.33	22.63	34.62
	8-shot	55.64	20.88	33.28	36.60
Qwen3-32B	4-shot	51.00	0.00	16.08	22.36
	6-shot	50.36	0.40	14.48	21.75
	8-shot	50.18	0.39	13.48	21.34
Qwen2.5-72B-Instruct	4-shot	49.62	38.07	2.78	30.16
	6-shot	39.48	46.84	5.67	30.67
	8-shot	29.45	48.95	17.96	32.12
Llama-3.3-70B-Instruct	4-shot	55.82	13.83	58.82	42.83
	6-shot	62.60	22.40	50.36	45.12
	8-shot	60.67	27.37	53.69	47.24
CoCoDet (Ours)	-	98.94	97.41	98.37	98.24

Table 13: Performance of LLMs with increasing numbers of in-context examples.

E.5 Confusion Matrix Analysis

To visually assess the performance of different models, we present the confusion matrices for LLM-based detectors and our ablation studies.

Figures 7 and 8 show the results for various LLM-based detectors in zero-shot and few-shot settings, respectively. These results highlight a general difficulty across models in correctly identifying reviews with mix content, which is often misclassified as AI. Figure 9 demonstrates the impact of each component in our CoCoDet model, where removing any single component leads to a noticeable degradation in performance.

E.5.1 LLM-based Detectors

An analysis of the confusion matrices for LLM-based detectors, reveals significant limitations in their ability to perform the ternary classification.

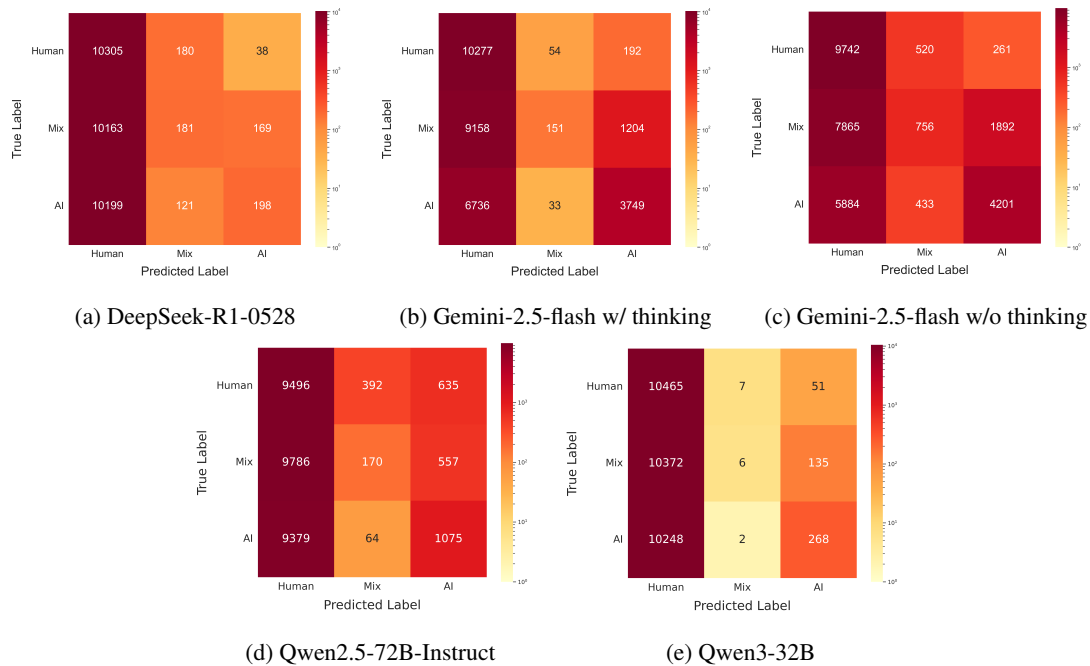


Figure 7: Confusion matrices for baseline LLM-based detectors in the zero-shot setting. The performance of five models is shown: (a) DeepSeek, (b) Gemini with CoT, (c) Gemini without CoT, (d) Qwen2.5, and (e) Qwen3.

In the zero-shot setting 7, performance is exceptionally poor, with most models failing to comprehend the core task. Most detectors exhibit an extreme classification bias, predicting nearly all inputs as human regardless of their true label. This suggests that without concrete examples, the models default to a simplistic classification mode rather than performing meaningful content analysis.

Performance improves notably in the few-shot setting 8, where most models demonstrate a basic understanding of the task’s structure. This is evidenced by a clear trend: as the true label shifts from human to AI, the number of human predictions (leftmost column) generally decreases while AI predictions (rightmost column) increase. However, a critical and universal weakness across both settings is the inability to correctly handle the mix category. These blended-authorship texts are consistently misclassified as either purely human or AI, with the diagonal values for the mix class remaining exceptionally low. This indicates that while few-shot prompting provides a basic sensitivity to the degree of AI involvement, these general-purpose models still lack the nuanced capability to disentangle content origin from style, making them unreliable for identifying challenging, real-world collaboration scenarios.

E.5.2 Ablation Study

The confusion matrices provide a visual analysis of the model’s failure modes when components are removed, revealing the specific error patterns each component is designed to prevent. The full CoCoDet model exhibits a clean diagonal with minimal errors, almost completely eliminating the critical misclassifications between ‘AI’ and ‘human’.

Removing the margin losses reveals their distinct roles. Ablating the Base Margin primarily increases confusion between adjacent classes, particularly causing more AI-written texts to be mistaken for mix. The removal of the Cost Margin is more detrimental, leading to a sharp increase in the most severe error type: AI-written texts being misclassified as human. This confirms that the Cost Margin is vital for preventing such critical failures.

Ablating the auxiliary tasks highlights their complementary contributions to feature disentanglement. Without the Content Source task, the model’s ability to ground predictions in substantive origin weakens, increasing confusion between the AI and mix categories. The removal of the Textual Style task results in a generalized performance degradation, with an observable increase in misclassifications across all true labels. This visually demonstrates that explicit supervision on style is crucial for overall robustness, reinforcing their

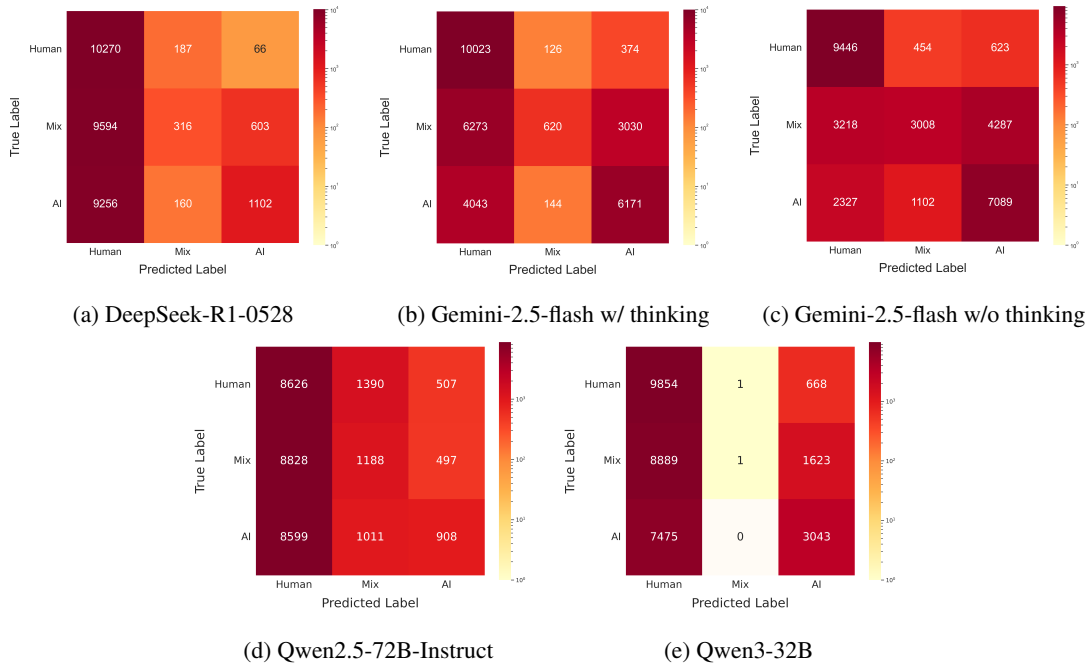


Figure 8: Confusion matrices for baseline LLM-based detectors in the few-shot setting. The performance of five models is shown: (a) DeepSeek, (b) Gemini with CoT, (c) Gemini without CoT, (d) Qwen2.5, and (e) Qwen3.

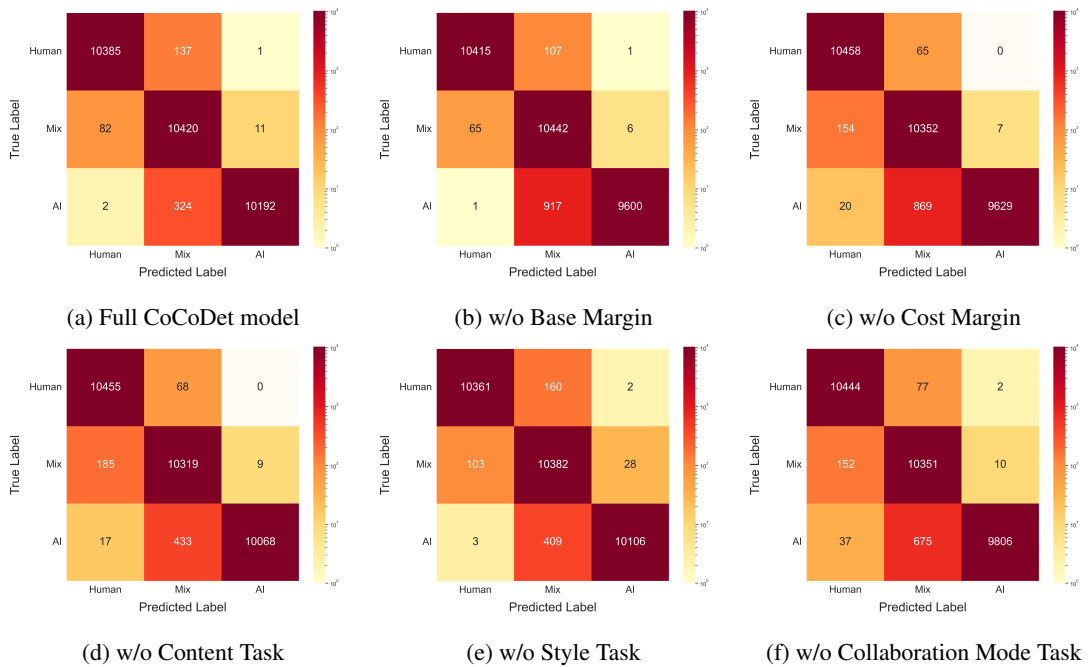


Figure 9: Confusion matrices for the ablation study of our CoCoDet model. (a) The full CoCoDet model. The other subfigures show performance after removing a specific component: (b) the base margin from CSM Loss, (c) the cost margin from CSM Loss, (d) the content attribution task, (e) the style attribution task, and (f) the collaboration mode task.

complementary nature. Finally, removing the Fine-grained Collaboration Mode Task causes the most widespread performance collapse, reflected in a substantial increase in both AI-to-mix errors and AI-to-human misclassifications.

E.6 Embedding Visualization

To provide a qualitative understanding of the feature space learned by CoCoDet, we visualize the embeddings for the main content-composition task using t-SNE. Figure 10 plots the 2D projection of the final hidden-layer representations for each sample in the test set, colored by its ground-truth label: ‘human’ (blue), ‘mix’ (orange), or ‘ai’ (red).

The visualization reveals a clear and meaningful structure. The ‘human’ and ‘ai’ classes form distinct, separated clusters, indicating that our model learns highly discriminative features. Crucially, the ‘mix’ class embeddings are not randomly scattered but form their own coherent clusters that bridge the ‘human’ and ‘ai’ regions. This demonstrates that CoCoDet recognizes mixed-authorship text as a distinct category rather than a simple interpolation. This well-structured feature space qualitatively validates our model’s ability to disentangle content provenance, corroborating the quantitative results.

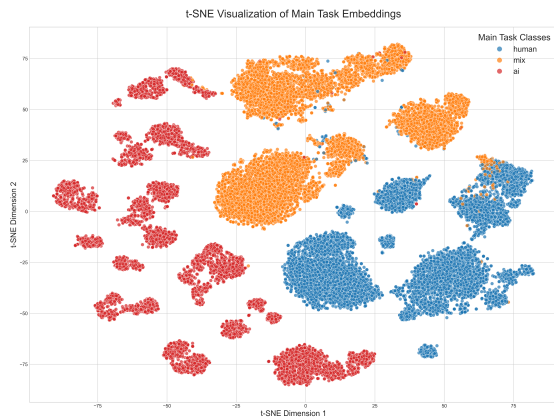


Figure 10: t-SNE visualization of the test set embeddings from CoCoDet. Samples are colored by their ground-truth content-composition class: Human (blue), Mix (orange), and AI (red). The clear clustering demonstrates the model’s ability to learn a discriminative feature space.

E.7 Analysis of Content Consistency After Semantic-Invariant Operations

To visually demonstrate how semantic-invariant operations impact the textual presentation of peer reviews while preserving their core substantive content, we present several case studies below. These

examples provide qualitative evidence supporting our data construction methodology, where operations like machine translation and polishing are treated as stylistic modifications that do not alter the fundamental content.

HW&MT As shown in Table 14, machine back-translation introduces subtle stylistic changes. For instance, “proves that” is rephrased as “demonstrates that”, and “lift the bound” becomes “improve the lower bound”. Similarly, “largely unexplored” is rephrased to the more formal “not been sufficiently explored”. Despite these variations in vocabulary and phrasing, the core scientific arguments, including the identified strengths (e.g., problem importance) and weaknesses (e.g., small training sets), remain faithfully preserved. This demonstrates that the translation operation primarily affects surface-level lexical choice without altering the review’s substantive evaluation.

HW&MP Table 15 illustrates the effect of machine polishing on a human-written review that contains grammatical errors and less formal phrasing. The original text’s “A couple questions I have” is transformed into the more structured “I have a few questions regarding the method”, and the questions are automatically enumerated. Phrasing like “I found the idea...original and interesting” is elevated to “The core idea is original and compelling”. Crucially, all the author’s original points are retained verbatim or with minor rephrasing. The polishing operation thus acts as a stylistic refiner, improving fluency and formality while maintaining the integrity of the human-authored content.

MG&MP This case study, presented in Table 16, highlights how polishing can significantly alter the style of an already well-formed, machine-generated text. While the original ‘MG’ version is grammatically correct, the polished version exhibits a more sophisticated narrative flow. For example, “This is a theoretically intriguing direction” is rephrased to the more assertive “This geometric perspective...is a compelling...direction”. The list of four identified issues is preserved in its entirety, but the surrounding sentence structures are varied to be less repetitive. The polishing operation effectively acts as a “humanizing” step, not by changing the information, but by improving the rhetorical style to be more cohesive and persuasive, thus demonstrating a stylistic shift while content remains constant.

HW	HW & MT
<p>This paper first proves that inter-class distinctiveness and intra-class compactness among hash codes determine the lower bound of hash codes' performance. And it shows that promoting these two characteristics could lift the bound and improve hash learning. Then it proposes a surrogate model to fully exploit such objective by estimating posterior of hash codes. Extensive experiments reveal effectiveness of the proposed method.</p>	<p>This paper first demonstrates that the inter-class distinguishability and intra-class compactness determine the lower bound of hash code performance. It also shows that promoting these two properties can improve the lower bound and enhance hash learning. Then, the paper proposes a proxy model that leverages this objective by estimating the posterior probability of hash codes. Extensive experiments reveal the effectiveness of the proposed method.</p>
<p>Strengths:</p> <ol style="list-style-type: none"> 1. The studied problem is interesting and important because a theoretical analysis on criteria of learning good hash codes remains largely unexplored. 2. The proposed method seems to be reasonable and effective. 3. Experiment seems to be extensive. 	<p>Strengths:</p> <ol style="list-style-type: none"> 1. The research question is interesting and important, as theoretical analysis of the criteria for learning good hash codes has not been sufficiently explored. 2. The proposed method appears reasonable and effective. 3. The experiments seem quite extensive.
<p>Weaknesses:</p> <ol style="list-style-type: none"> 1. There exist some typos and grammatical errors in the paper. 2. The training sets on all datasets are relatively small. 	<p>Weaknesses:</p> <ol style="list-style-type: none"> 1. There are some spelling and grammatical errors in the paper. 2. The training sets for all datasets are relatively small.
<p>Is the performance improvement still significant when we increase the size of training set? The authors have adequately addressed the limitations and potential negative societal impact of their work.</p>	<p>Does the performance improvement remain significant when the size of the training set is increased? The authors adequately address the limitations and potential negative societal impacts of their work.</p>

Table 14: Content consistency in machine translation (HW & MT)

E.8 Error Analysis

Analysis of LLM reasoning processes To investigate the failure modes of Large Language Models (LLMs) as detectors, we examine their Chain-of-Thought (CoT) reasoning. Our analysis reveals a consistent limitation: LLMs tend to perform a surface structural assessment of the text rather than a deep analysis of its substantive content.

The reasoning processes of Qwen3 and DeepSeek, shown below, exemplify this behavior. Both models begin by correctly identifying the document's structure and topic, for instance, recognizing a review's "Strengths" and "Weaknesses" sections or its subject matter. This initial step resembles a form of content analysis.

However, the models' subsequent reasoning for their final classification pivots to stylistic and qualitative evaluations. They base their decision on perceived attributes such as "technical language",

"nuanced points", or "natural flow". In essence, the models equate the successful imitation of an expert's writing style and structure with genuine human authorship. They do not demonstrate an ability to question the provenance of the well-formed arguments themselves.

This finding suggests that prompt-based LLM detectors primarily operate by matching text to a learned schema of what an expert review should look like, rather than by performing true content-based source attribution. This reliance on form and style, rather than substance, makes them unreliable for distinguishing high-quality AI-generated text from human writing.

An Error Case of the CoT of Qwen3

Okay, let's tackle this. The user wants me to determine if the given text is human, AI,

HW

This paper proposed a new black-box attack method in the hard-label setting. By using a well-designed policy network in a novel reinforcement learning formulation, the new method learns promising search directions of the adversarial examples and showed that query complexity is significantly reduced in experiments.

A couple questions I have:

How are β_1 and β_2 chosen?

Is there any convergence analysis? How do we guarantee the proposed attacking method will converge?

The proposed method is introduced as an attack that minimize L_2 distance. Is it possible to extend this attacking method to L_{inf} ?

I am curious, have you tried evaluate the attacking methods on DenseNet as the victim model?

Overall, I think this paper is very readable and is clearly written with a very good background and context. I found the idea of the paper original and interesting. And the authors have conducted experiments that show their new method has the best query efficiency, which is reasonable and aligns with their idea. For cons, this paper does not have a convergence analysis. And if the experiments could be conducted on more data sets and more victim models, then it would be more convincing.

HW & MP

This paper introduces a novel black-box attack method operating in the hard-label setting. The core contribution is a reinforcement learning formulation that employs a meticulously designed policy network to learn promising search directions for adversarial examples. This approach demonstrably reduces query complexity, as evidenced by experimental results.

I have a few questions regarding the method:

1. How are the hyperparameters β_1 and β_2 determined?

2. Is there any convergence analysis for the proposed attack method? How can we guarantee its convergence?

3. The proposed method is presented as an L2-norm-minimizing attack. Is it possible to extend this approach to the L-infinity norm setting?

4. Have the authors evaluated the attack method against DenseNet as a victim model?

Overall, the paper is well-written, clear, and provides excellent background and context. The core idea is original and compelling. The authors have conducted experiments that effectively demonstrate the superior query efficiency of their new method, which aligns well with the proposed concept. On the downside, the paper lacks a convergence analysis. Furthermore, the experimental results would be more convincing if evaluated on a broader range of datasets and victim models.

Table 15: Content consistency in machine polishing (HW & MP)

or a mix. The examples provided show that “human” is when the core is human-written, even if edited by AI. “AI” is when the core is AI-generated, even if a human polished it. “Mix” is when content is from human and AI. Looking at the text in question: it’s a detailed review of a paper on reward imputation in contextual batched bandits. The review discusses the method, analysis, experiments, and suggests improvements. The language is technical, with specific references to theoretical guarantees, experimental results, and comparisons.

The structure is formal and thorough, which is typical in academic reviews. However, AI-generated reviews might have a more pol-

ished or generic tone, but this one has specific critiques like suggesting comparisons with other techniques, hyperparameter sensitivity, and applicability to other settings. These are nuanced points that a human reviewer might note based on their expertise. Also, the text includes phrases like “the paper is well-crafted” and “the claims seem accurate”, which are subjective judgments a human might make. AI might generate such a review, but the depth and specificity here lean more towards a human author. There’s no clear indication of distinct AI-generated sections mixed with human parts.

Therefore, I think this is a human-written review. The detailed, critical analysis and

specific suggestions are hallmarks of a human reviewer’s input.

An Error Case of the CoT of DeepSeek-R1

Alright, I need to classify this text as ‘human’, ‘ai’, or ‘mix’ based on the given rules. Let me start by reading the content carefully. The text is a review of a research paper submission. It discusses strengths, weaknesses, and suggestions for improvement. The structure is organized with clear headings like “Strengths”, “Weaknesses”, and “Questions and Suggestions for Improvement”. The language is critical but constructive, pointing out areas where the paper lacks detail, such as empirical evaluations and mathematical formulations.

Looking at the examples provided, the ‘human’ example was detailed with specific critiques and personal observations. The ‘ai’ example was more structured and used bullet points but still had a formal tone. The ‘mix’ example had sections where parts seemed human-written and others AI-generated.

Here, the reviewer uses phrases like “The paper could benefit from” and “The authors should consider...” which suggest personal judgment. The questions raised are specific, like handling high-dimensional settings, which indicates deep understanding. The suggestions for including empirical evaluations and clearer structure show critical thinking typical of a human reviewer. There’s no obvious sign of AI-generated content, like overly formal or generic statements. The entire text flows naturally without sections that stand out as AI-generated.

Error mode of CoCoDet An analysis of the confusion matrix in Figure 9a reveals insightful patterns regarding the inherent challenges of this detection task. The predominant form of misclassification involves a slight confusion at the boundaries of the “Mix” class, where purely “Human” or “AI” samples are occasionally categorized as “Mix”. This indicates that CoCoDet has successfully learned to distinguish the core characteristics of purely human and AI-authored content, and its remaining errors are confined to the most ambiguous cases of human-AI interaction. For instance,

some misclassifications occur with high-quality AI-generated text that perfectly emulates human stylistic nuances, or with human-written text that has undergone minor, algorithmically-induced stylistic alterations. In these challenging scenarios, where the line between content preservation and content contribution blurs, the model conservatively classifies the text as “Mix”. This behavior, while technically an error, can be interpreted as a failsafe mechanism, avoiding a more severe misclassification (e.g., flagging polished Human text as purely AI). A secondary, less frequent error pattern is the misclassification of “Mix” samples as either purely “Human” or “AI”. This typically occurs when the contribution from the minority source is exceptionally subtle. Crucially, our Cost-Sensitive Margin Loss proves highly effective in preventing the most critical errors (direct confusion between the “Human” and “AI” classes). The residual errors are therefore concentrated at the fuzzy edges of collaboration, highlighting the task’s difficulty rather than a limitation of our content-centric approach.

Case Study of Severe Errors To provide a more intuitive understanding of the remaining challenges and failure modes, we present two representative severe error cases of CoCoDet. For brevity and clarity, the reviews have been slightly shortened.

Human-Written Review Misclassified as AI

This paper presents a straightforward approach for creating neural networks that can operate under varying resource constraints, offering a trade-off between computational cost and solution quality. The key idea is to share filters between networks of different sizes, with the smallest network having the cheapest filters, and train them jointly with weight sharing. A crucial practical observation is that batch normalization parameters should not be shared across filters to achieve good results. The most surprising and novel finding of this work is that even the highest-quality vision networks benefit substantially from this joint training methodology, outperforming their solo training counterparts without weight sharing, particularly for object detection and segmentation tasks, although not for recognition. This unexpected result introduces a new and unanticipated method

for training improved segmentation models. Notably, the paper acknowledges the lack of understanding behind this phenomenon and leaves its explanation to future work, opening up a promising avenue for further research and exploration. Overall, this line of inquiry has the potential to yield interesting and impactful results.

the distribution of professional human writing.

AI Review Misclassified as Human

The paper investigates [Topic omitted]. The authors introduce Helper-based Adversarial Training (HAT), which incorporates wrongly labelled examples to [Task omitted]. The work is well-motivated and presents [...] The analysis of margin dynamics during adversarial training is insightful and aligns with prior work (e.g., Zhang et al., 2019) [...] The authors should clarify how HAT differs conceptually from these methods and whether the 'wrongly labelled examples' are generated via PGD-like procedures or sampled randomly. The experimental evaluation is a strength, but the paper could be strengthened by (1) comparing against state-of-the-art robust training methods (e.g., FixFree...). Additionally, the paper should address how the margin definition (Equation 2) relates to existing robustness metrics. Further ablation studies on the number and selection of helper examples would strengthen the contribution. Overall, the paper makes a valuable empirical contribution to understanding the accuracy-robustness trade-off.

The Human Case (from the hwmp category) is misclassified as AI primarily due to its shallow content. This review lacks specific data or results from the paper. Because it relies heavily on standard academic phrasing, its content closely resembles typical AI outputs. This reveals that distinguishing highly generic human-written summaries from AI-generated ones remains a persistent challenge.

The AI Case (from the mg category) successfully bypasses detection because its composition is highly characteristic of expert human reviews. By incorporating numerous technical references, specific citations (e.g., Equation 2), and actionable peer-review suggestions, the model (Qwen3) generates a grounded content profile that aligns with

MG

The paper addresses the gradient mismatch problem in training Quantized Neural Networks (QNNs) via a novel approach called ManiQuant, which leverages Riemannian manifold learning to revise the Straight-Through Estimator (STE). The core idea is to model the parameter space as a metric tensor in a Riemannian manifold, enabling a more accurate gradient approximation through the non-differentiable quantization function. This is a theoretically intriguing direction, as it introduces geometric priors into gradient estimation, which has been relatively underexplored in the QNN literature. The experimental results on CIFAR10/100 and ImageNet suggest improved stability and performance compared to standard STE, supported by ablation studies. However, the review identifies several key issues. First, the technical details of how the Riemannian metric tensor is defined and computed remain under-specified. For instance, it is unclear whether the metric is learned during training or derived from a fixed geometric structure (e.g., based on curvature or data distribution). This lack of clarity weakens the reproducibility and interpretability of the method. Second, the paper primarily benchmarks against STE-based methods but does not compare with non-STE alternatives like differentiable quantization layers or optimization-based quantization frameworks. A more comprehensive comparison with existing QNN strategies would strengthen the evaluation. Third, while the results on standard image datasets are promising, extending experiments to more challenging tasks (e.g., video, NLP) or larger-scale models (e.g., ViT, ResNet-152) would better demonstrate the method's generalizability. Finally, the theoretical justification for why Riemannian embeddings reduce gradient mismatch is not fully developed. A derivation linking the manifold structure to gradient stability would enhance the contribution. Overall, this work offers a novel geometric perspective on STE but requires deeper technical exposition and broader empirical validation to establish its significance for the QNN community.

MG & MP

This paper proposes a fresh approach to addressing the gradient mismatch issue in Quantized Neural Networks (QNNs) through ManiQuant, a method that utilizes Riemannian manifold learning to refine the Straight-Through Estimator (STE). By treating the parameter space as a Riemannian manifold with a metric tensor, ManiQuant achieves a more precise gradient approximation, even in the face of non-differentiable quantization functions. This geometric perspective on gradient estimation is a compelling, albeit underexplored, direction in QNN research. Experimental results on benchmark datasets such as CIFAR10/100 and ImageNet indicate that ManiQuant offers improved stability and performance over traditional STE methods, with supporting evidence from ablation studies. However, several concerns need to be addressed. The definition and computation of the Riemannian metric tensor, for instance, are not thoroughly explained, leaving unclear whether it's learned during training or predetermined by a specific geometric structure. This omission hinders the method's reproducibility and interpretability. Furthermore, the comparison to other quantization methods is limited, primarily focusing on STE-based approaches while neglecting alternative strategies such as differentiable quantization layers or optimization-based frameworks. A more inclusive comparison would significantly enhance the evaluation. Additionally, while the current results are promising, they are based on standard image datasets, and extending the experiments to more complex tasks or larger models would provide a more robust demonstration of the method's versatility. Lastly, a detailed theoretical explanation of how Riemannian embeddings mitigate gradient mismatch is lacking. Developing this justification would substantially bolster the paper's contribution. In essence, ManiQuant introduces an innovative geometric viewpoint to STE, but to firmly establish its impact on the QNN community, it requires more comprehensive technical details and empirical validations.

Table 16: Content consistency in machine polishing (MG & MP)