The Feasibility of Topic-Based Watermarking on Academic Peer Reviews

Anonymous ACL submission

Abstract

Large language models (LLMs) are increasingly integrated into academic workflows, with many conferences and journals permitting their use for tasks such as language refinement and literature summarization. However, their use in peer review remains prohibited due to concerns around confidentiality breaches, hallucinated content, and inconsistent evaluations. As LLM-generated text becomes more indistinguishable from human writing, there is a growing need for reliable attribution mechanisms to preserve the integrity of the review process. In this work, we evaluate topic-based watermarking (TBW), a semantic-aware technique designed to embed detectable signals into LLM-generated text. We conduct a systematic assessment across multiple LLM configurations, including base, few-shot, and fine-tuned variants, using authentic peer review data from academic conferences. Our results show that TBW maintains review quality relative to non-watermarked outputs, while demonstrating robust detection performance under paraphrasing. These findings highlight the viability of TBW as a minimally intrusive and practical solution for LLM attribution in peer review settings.

1 Introduction

As large language models (LLMs) continue to evolve, their adoption has accelerated in academic writing (Dergaa et al., 2023; Editorials, 2023). LLMs are widely used for language polishing, literature search, and low-novelty writing. Many conferences now explicitly allow authors to use LLMs for certain tasks, provided that authors retain full responsibility for the content (ACL, 2025a; NeurIPS, 2025; ICML, 2025a). These policies uphold pre-LLM expectations around authorship and accountability while adapting to new technological norms.

In contrast, the use of LLMs by peer reviewers is widely prohibited (ACL, 2025b; NeurIPS, 2025; ICML, 2025b). Such practices risk confidentiality breaches, low-quality evaluations, and data exposure (Zhou et al., 2024; Maini et al., 2024). Recent empirical studies suggest, however, that LLM-assisted reviews are already present in major conferences, leading to inflated scores, reduced reviewer confidence, and distortions in paper rankings (Liang et al., 2024; Latona et al., 2024; Ye et al., 2024). These developments underscore the urgency of developing attribution mechanisms to detect and manage LLM usage in peer review.

Distinguishing between machine- and human-authored reviews has become difficult, as LLM-generated content continues to improve. This creates an urgent need for technical mechanisms to trace review provenance. *Watermarking* offers a promising approach, embedding imperceptible, machine-detectable signatures into generated text (Zhao et al., 2024). However, existing work focuses on general-domain text, with limited analysis in peer review contexts (Liu et al., 2024; Zhao et al., 2023).

In this paper, we present the first focused evaluation of topic-based watermarking in the context of academic peer reviews. Topic-based watermarking (TBW) offers a balance of efficiency, robustness to paraphrasing, and minimal impact on generation quality, making it suitable for peer review, where stylistic fidelity and semantic coherence are critical. It also supports domain adaptation through customizable topic lists aligned with peer review structure. Moreover, TBW relies on a topic-matching assumption that naturally holds in this setting, where reviews are expected to stay aligned with the subject of the paper.

Our goal is to assess whether TBW can preserve review quality and semantic fidelity while offering attribution under realistic adversarial settings. We evaluate across three LLM configurations: a pretrained base model, a few-shot configuration, and a fine-tuned model using authentic reviews from AI and ML conferences. Our analysis examines generation and semantic quality, robustness to paraphrasing, and classifier-based attribution. We further compare TBW against general-purpose watermarkings and find that TBW offers better preservation of text quality, highlighting its suitability for domain-sensitive tasks like peer review.

Without effective attribution mechanisms, the credibility and rigor of academic conferences could erode, leading to lower-quality evaluations and increased reliance on potentially unverifiable, machine-generated feedback. Watermarking provides a practical and minimally disruptive approach for LLM accountability, helping to safeguard academic standards while accommodating the evolving role of generative models. Related work on LLM watermarking and detection approaches is provided in §A.

2 Topic-Based Watermarks

Topic-based watermarking (TBW) (Anonymous, 2024) is a semantic-aware watermark that subtly influences a language model's token selection process to leave a detectable signature. Unlike earlier schemes such as KGW (Kirchenbauer et al., 2023), which rely on randomly partitioned vocabularies, TBW constructs topic-specific token subsets ("green lists") aligned with the semantic content of the input prompt, helping preserve fluency while enhancing robustness against paraphrasing.

Token-to-Topic Mappings. TBW first assigns tokens to topic-specific green lists using semantic similarity. A small set of topics $t_1, \ldots, t_K$ is defined, each represented by an embedding $\mathbf{e}_{t_i}$ computed via a sentence embedding model. Each token $v \in V$ in the model's vocabulary is embedded as $\mathbf{e}_v$, and its cosine similarity with each topic embedding is computed. If the maximum similarity exceeds a threshold τ, the token is assigned to the green list G_{t_i} for the most similar topic. Tokens that do not meet this threshold are placed in a residual set and evenly distributed across all green lists to maintain full vocabulary coverage.

Generation & Detection. During generation, the most relevant topic is identified from the input prompt using keyword extraction, and TBW adds a small logit bias δ to all tokens in the corresponding green list. This increases the likelihood of sampling topic-aligned tokens while preserving the model architecture and generation efficiency. The watermark strength is controlled by δ: higher values produce stronger attribution signals but may cause detectable shifts in word choice.

For detection, TBW recovers the relevant topic from the input text and counts green-list tokens g relative to total tokens n. A z-score quantifies whether the green-token rate exceeds an expected baseline proportion γ:

$$z = \frac{g - \gamma \cdot n}{\sqrt{n \cdot \gamma \cdot (1-\gamma)}}.$$

If $z > z_{\text{threshold}}$, the text is classified as watermarked.

Rationale for TBW. We select TBW for its combination of robustness, adaptability, and minimal overhead (Anonymous, 2024). TBW is particularly well-suited to peer review contexts where paraphrasing represents a realistic threat model where reviewers may rephrase generated content to improve clarity or avoid detection, but are unlikely to introduce noise that would degrade review quality. Additionally, TBW's requirement of topic alignment between the input prompt and generated text naturally holds in peer review, where content must remain semantically aligned with the paper under evaluation. This contrasts with open-ended generation tasks where topic drift can undermine detection effectiveness. Finally, TBW's semantic biasing strategy preserves fluency and style while supporting domain adaptation through customizable topic sets $t_1, \ldots, t_K$ that can be tailored to specific research fields or venues.

3 Experimental Setup

We simulate realistic LLM-based peer review generation by training and prompting models to write reviews conditioned on a paper's title and abstract. We condition on paper abstracts rather than full text due to context length constraints and structured data availability.

3.1 Dataset

We construct a dataset of paper titles, abstracts, and corresponding reviews from ICLR and NeurIPS conferences using the OpenReview API (OpenReview, 2024)[1]. To minimize risk of including LLM-generated content, we restrict data to conferences before ChatGPT's release (November 2022) (OpenAI, 2022): ICLR 2018–2023 and NeurIPS 2021–2022.

[1] Code and dataset available at https://anonymous.4open.science/r/Watermarking-LLM-Conference-E051/.

The final dataset contains approximately 19,000 reviews, each including summary, strengths/weaknesses, and recommendation scores. For each paper, we randomly sample a single review to construct prompt-completion pairs, ensuring reviewer diversity while avoiding overrepresentation. Detailed statistics are in §C.1.

3.2 TBW Domain Adaptation

We adapt TBW to peer review by modifying the topic sets while retaining original parameter settings. Instead of general-purpose topics (e.g., `technology`, `sports`), we define domain-specific topics: {`theory`, `applications`, `models`, `optimization`} to capture themes in ML conference reviews. We use logit bias $\delta = 2.0$ and similarity threshold $\tau = 0.7$ for primary experiments, with additional evaluation at $\tau = 0.3$. Complete parameter details are provided in §C.2.

3.3 Model Configurations

To assess the feasibility of TBW across varying levels of model adaptation and reviewer effort, we utilize the `Llama-3.1-8B` (Grattafiori et al., 2024) in three configurations: base, few-shot, and fine-tuned. The base configuration uses the pretrained model without additional training or prompt engineering, simulating minimal reviewer effort. The few-shot setting provides the model with example peer reviews as part of the input prompt, enabling it to replicate the expected format and tone with lightweight guidance. The fine-tuned configuration involves supervised training on peer review data using parameter-efficient methods, resulting in a model more aligned with the review-writing task. This model size offers a practical balance between computational efficiency and generation quality suitable for our multi-configuration experiments. Detailed setup parameters for few-shot prompting and fine-tuning are provided in §C.3.

4 Experiments

We conduct a series of experiments across multiple dimensions, including text quality, robustness to paraphrasing, and classifier-based attribution.

4.1 Generation Quality

To assess TBW's impact on generation quality, we evaluate perplexity across 1,000 samples per model configuration (base, few-shot, fine-tuned), comparing against unwatermarked baselines and two existing schemes: KGW (Kirchenbauer et al., 2023) and Google's SynthID-Text (Dathathri et al., 2024). Each sample contains approximately 200 ± 5 tokens. Parameter configurations, additional threshold results, and BERTScore evaluations are detailed in §D.

We compute perplexity using the generating model (`Llama-3.1-8B`) as a fluency proxy, where lower values indicate higher naturalness. For visualization clarity, values above 20 are truncated in Figure 1. The results demonstrate that TBW introduces minimal perplexity degradation while maintaining fluency compared to existing watermarking schemes. Notably, over 50% of unwatermarked base model generations exceed perplexity 20, whereas nearly all TBW outputs remain below this threshold, the best performance among all watermarking methods tested (detailed counts in §D.2). This suggests TBW not only preserves generation naturalness but enhances lexical consistency in low-context scenarios by steering generation toward topic-relevant vocabulary.

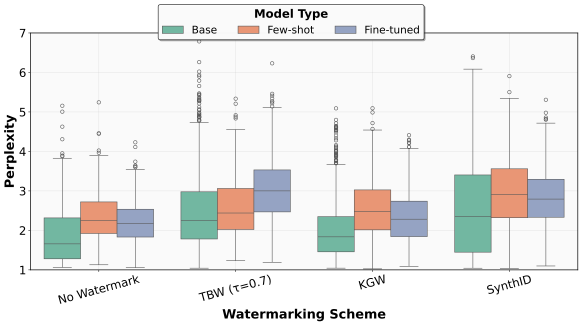


Figure 1: Perplexity distributions across model and watermark configurations. Lower values indicate better fluency. Values above 20 are truncated for clarity.

4.2 Robustness to Paraphrasing Attacks

We evaluate TBW's resilience against paraphrasing attacks, which represent a realistic threat model where reviewers may rephrase LLM-generated content to evade detection while preserving semantic meaning. We focus on full-text paraphrasing attacks that best reflect plausible reviewer behavior, excluding token-level or partial edit scenarios.

We generate 1,000 samples per model configuration (base, few-shot, fine-tuned) for each watermarking scheme, with each sample containing 200 ± 5 tokens. Paraphrasing attacks are implemented using two established models: PEGASUS (Zhang et al., 2020) and DIPPER (Krishna et al., 2023) (configured with `lexical=60` and

Language Model	Attacks	Detection Accurcy TBW	 KGW	 SynthID
Base	No Attack	0.946	**0.971**	0.909
	PEGASUS	**0.847**	0.477	0.135
	DIPPER	**0.876**	0.754	0.173
Few-shot	No Attack	0.622	**0.975**	0.959
	PEGASUS	**0.580**	**0.580**	0.359
	DIPPER	0.517	**0.748**	0.225
Fine-tuned	No Attack	0.880	0.926	**0.960**
	PEGASUS	**0.583**	0.437	0.180
	DIPPER	0.584	**0.657**	0.159

Table 1: Detection accuracy of TBW, KGW, and SynthID across model configurations and paraphrasing attack types. Scores reflect the proportion of correctly identified watermarked samples per condition.

Metric	Base NW	 TBW	Few-shot NW	 TBW	Fine-tuned NW	 TBW
			BERT			
Accuracy	0.290	**0.321**	0.403	**0.437**	0.400	**0.416**
Precision	**0.353**	0.346	**0.373**	0.366	**0.367**	0.366
Recall	0.328	**0.342**	**0.379**	0.369	**0.370**	0.367
F1	0.278	**0.317**	**0.360**	0.358	0.364	**0.366**
			RoBERTa			
Accuracy	**0.486**	0.432	0.399	**0.424**	0.406	**0.443**
Precision	0.344	**0.357**	0.362	**0.371**	0.367	**0.401**
Recall	0.341	**0.352**	0.368	**0.371**	0.374	**0.403**
F1	0.305	**0.350**	0.337	**0.353**	0.367	**0.402**

Table 2: Overall classification performance on original LLM-generated reviews. Metrics are averaged over Accept, Borderline, and Reject classes.

order=40), following standard robustness evaluation protocols (Hou et al., 2024; Liu and Bu, 2024).

Table 1 shows detection accuracy across three attack conditions. TBW demonstrates superior robustness compared to existing methods, maintaining detection accuracy for base models and fine-tuned models even under paraphrasing attacks. In contrast, KGW and SynthID show degradation, with SynthID dropping to as low as 13.5% accuracy under PEGASUS attacks. The few-shot configuration exhibits reduced performance across all methods, with TBW achieving 58% accuracy under PEGASUS and 52% under DIPPER. This degradation likely stems from topic misalignment between the prompt examples and target papers, which weakens topic alignment and reduces detectability post-paraphrasing.

We verify that TBW maintains low false positive rates on human-written reviews through its vocabulary partitioning strategy, which preserves lexical diversity across topic-specific green lists. Complete ROC analysis is provided in §E.

4.3 Classifier-Based Attribution

To complement watermark detection, we evaluate whether LLM-generated peer reviews can be attributed to their original review labels (accept, borderline, reject) using transformer-based classification models. This task assesses whether watermarking affects the semantic structure necessary for accurate label prediction in downstream interpretation tasks.

We use a balanced dataset of 9,000 training samples (3,000 per class) and 1,000 test samples. Reviews are mapped to three categories based on original scores: 1–4 (reject), 5–6 (borderline), and 7–10 (accept). We train BERT and RoBERTa classifiers to predict review ratings based on generated review text, evaluating performance on both watermarked and non-watermarked reviews.

As shown in Table 2, TBW demonstrates mixed but generally positive effects on classification performance. While some configurations show modest degradation (e.g., RoBERTa base accuracy), the majority of results indicate that TBW causes little to no degradation and often leads to improvements in both accuracy and F1 scores. We hypothesize that this occurs because topic-based watermarking encourages more topically consistent language that aligns better with the underlying review content.

These findings reinforce TBW's suitability for domain-sensitive contexts like peer review, where both traceability and semantic fidelity are critical. Additional experimental details, including class-specific performance metrics, evaluation under lower similarity thresholds ($\tau = 0.3$), and robustness evaluation under paraphrasing attacks, are provided in §F.

5 Conclusion

We present the first systematic evaluation of topic-based watermarking (TBW) for academic peer review, demonstrating that TBW preserves generation quality while maintaining robust detection under paraphrasing attacks. Our experiments across base, few-shot, and fine-tuned LLM configurations show that TBW's semantic grounding naturally aligns with peer review constraints, where content must remain topically consistent with the evaluated paper. These findings highlight TBW's viability as a minimally intrusive solution for LLM attribution in peer review settings, offering a practical mechanism to safeguard academic evaluation integrity. Discussion of deployment considerations and limitations is provided in §B.

Limitations

This work inherits a key limitation of topic-based watermarking: the topic-matching assumption. As noted in the original proposal (Anonymous, 2024), watermark detection may degrade if the semantic topic of the generated output drifts significantly from the original prompt. This is particularly challenging in open-domain generation, where the input prompt is often unavailable at detection time. However, in the context of peer review, this limitation is largely mitigated. Reviewers must prompt the LLM using the content of the paper, either by directly including the text or referencing its abstract and title, ensuring that the generated review remains topically aligned with the source. Furthermore, during detection, conference organizers have access to the submission itself, allowing them to reliably identify the intended topic and recover the correct green list. As a result, the topic-matching assumption holds in this use case.

A second limitation concerns deployment and coverage. For watermarking to serve as a reliable attribution mechanism, it must be consistently applied across all LLMs used in a given environment. This is a general challenge for watermarking approaches and not unique to TBW. If only certain LLM providers implement watermarking while others do not, users can simply switch to unwatermarked systems to bypass attribution. While the governance and policy mechanisms required to address this challenge are beyond the scope of this paper, we acknowledge that the effectiveness of TBW in real-world enforcement depends on broader coordination across providers and platforms.

Ethical Considerations

This work addresses the growing concern of unauthorized LLM usage in academic peer review. While many conferences permit LLM use for authoring papers, they explicitly prohibit it for generating reviews, citing risks to confidentiality, fairness, and accountability. Our goal is not to penalize reviewers but to support conference organizers in enforcing existing policies through lightweight and interpretable attribution tools. Topic-based watermarking introduces no additional risk to authors or reviewers, as it operates at the generation level without modifying model internals or relying on invasive detection mechanisms. We advocate for transparent disclosure of LLM usage in reviews and emphasize that attribution tools should be deployed with clear governance structures and ethical oversight.

References

ACL. 2025a. Acl rolling review call for papers. https://aclrollingreview.org/cfp#long-papers. Accessed: 2025-05-15.

ACL. 2025b. Arr reviewer guidelines. https://aclrollingreview.org/reviewerguidelines. Accessed: 2025-05-15.

Anonymous. 2024. Topic-based watermarks for large language models. *arXiv preprint arXiv:2404.02138*.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, Jamie Hayes, Nidhi Vyas, Majd Al Merey, Jonah Brown-Cohen, Rudy Bunel, Borja Balle, Taylan Cemgil, Zahra Ahmed, Kitty Stacpoole, and 5 others. 2024. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823.

Ismail Dergaa, Karim Chamari, Piotr Zmijewski, and Helmi Ben Saad. 2023. From human writing to artificial intelligence generated text: examining the prospects and potential threats of chatgpt in academic writing. *Biology of sport*, 40(2):615–622.

Nature Editorials. 2023. Tools such as chatgpt threaten transparent science; here are our ground rules for their use. *Nature*, 613(7945):612.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Abe Hou, Jingyu Zhang, Tianxing He, Yichen Wang, Yung-Sung Chuang, Hongwei Wang, Lingfeng Shen, Benjamin Van Durme, Daniel Khashabi, and Yulia Tsvetkov. 2024. SemStamp: A semantic watermark with paraphrastic robustness for text generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4067–4082, Mexico City, Mexico. Association for Computational Linguistics.

ICML. 2025a. Icml 2025 call for papers. https://icml.cc/Conferences/2025/CallForPapers. Accessed: 2025-05-15.

ICML. 2025b. Icml 2025 reviewer instructions. https://icml.cc/Conferences/2025/ReviewerInstructions. Accessed: 2025-05-15.

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

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

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 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 944–953.

Giuseppe Russo Latona, Manoel Horta Ribeiro, Tim R Davidson, Veniamin Veselovsky, and Robert West. 2024. The ai review lottery: Widespread ai-assisted peer reviews boost paper scores and acceptance rates. *arXiv preprint arXiv:2405.02150*.

Zhi-Qiang Li, Hui-Lin Xu, Hui-Juan Cao, Zhao-Lan Liu, Yu-Tong Fei, and Jian-Ping Liu. 2024. Use of artificial intelligence in peer review among top 100 medical journals. *JAMA Network Open*, 7(12):e2448609–e2448609.

Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, and 1 others. 2024. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on ai conference peer reviews. *arXiv preprint arXiv:2403.07183*.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. 2024. A semantic invariant robust watermark for large language models. In *The Twelfth International Conference on Learning Representations*.

Yepeng Liu and Yuheng Bu. 2024. Adaptive text watermark for large language models. *arXiv preprint arXiv:2401.13927*.

Pratyush Maini, Hengrui Jia, Nicolas Papernot, and Adam Dziedzic. 2024. Llm dataset inference: Did you train on my dataset? *Advances in Neural Information Processing Systems*, 37:124069–124092.

NeurIPS. 2025. Neurips 2025 policy on the use of large language models. `https://neurips.cc/Conferences/2025/LLM`. Accessed: 2025-05-15.

OpenAI. 2022. Introducing chatgpt. Accessed: 2025-05-16.

OpenReview. 2024. Openreview documentation. `https://docs.openreview.net/getting-started/using-the-api`. Accessed: 2025-05-16.

Leyi Pan, Aiwei Liu, Zhiwei He, Zitian Gao, Xuandong Zhao, Yijian Lu, Binglin Zhou, Shuliang Liu, Xuming Hu, Lijie Wen, Irwin King, and Philip S. Yu. 2024. MarkLLM: An open-source toolkit for LLM watermarking. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 61–71, Miami, Florida, USA. Association for Computational Linguistics.

Vishisht Rao, Aounon Kumar, Himabindu Lakkaraju, and Nihar B Shah. 2025. Detecting llm-written peer reviews. *arXiv preprint arXiv:2503.15772*.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2412.01708*.

Sungduk Yu, Man Luo, Avinash Madusu, Vasudev Lal, and Phillip Howard. 2025. Is your paper being reviewed by an llm? a new benchmark dataset and approach for detecting ai text in peer review. *arXiv preprint arXiv:2502.19614*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International conference on machine learning*, pages 11328–11339. PMLR.

Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. 2023. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*.

Xuandong Zhao, Sam Gunn, Miranda Christ, Jaiden Fairoze, Andres Fabrega, Nicholas Carlini, Sanjam Garg, Sanghyun Hong, Milad Nasr, Florian Tramer, and 1 others. 2024. Sok: Watermarking for ai-generated content. *arXiv preprint arXiv:2411.18479*.

Ruiyang Zhou, Lu Chen, and Kai Yu. 2024. 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.

A Related Work

Since the release of ChatGPT, LLMs have been rapidly adopted across various stages of the academic workflow. Their use has raised concerns about authorship and peer review integrity. Most conferences and journals now permit authors to leverage LLMs; however, this permissive stance does not extend to peer reviewers. Leading venues such as NeurIPS and ACL explicitly prohibit the use of LLMs by reviewers (NeurIPS, 2025; ACL, 2025b). These policies reflect growing concerns around review quality, including the risk of shallow or hallucinated feedback, reduced technical depth, and breaches of confidentiality that would compromise the double-blind review process (Li et al., 2024).

Despite these restrictions, recent studies suggest that LLM-assisted reviews are already present at major conferences. Liang et al. (2024) estimate that 5–15% of reviews were substantially modified using LLMs, with affected reviewers showing lower confidence and less engagement during rebuttals. Latona et al. (2024) report similar trends and observe a score inflation effect, while Ye et al. (2024) show that even subtle LLM manipulations can shift paper rankings. Together, these findings underscore the risks unauthorized LLM use poses to peer review fairness and rigor.

Given the increasing use of LLMs for peer review generation, recent work has focused on detecting and attributing such content. Much of this research explores classifier-based detection or semantic similarity methods aimed at identifying AI-generated text. For example, Yu et al. (2025) propose a detection method based on the semantic similarity between a known LLM-generated review and a test review, flagging a review as machine-generated when similarity exceeds a threshold. Similarly, Kumar et al. (2025) introduce a partition-based method under the assumption that a review contains both human- and LLM-written components. They segment the review into distinct points, complete each segment with a reference LLM, and measure semantic similarity between these completions and the original text to detect potential LLM involvement.

However, these detection methods fail under paraphrasing or hybrid-review scenarios, where even minor edits or partial human rewriting can evade detection. To address this limitation, watermarking offers a promising alternative by embedding identifiable signals directly into the generated text. One foundational method is the KGW algorithm (Kirchenbauer et al., 2023), which partitions the model's vocabulary into "green" and "red" token sets. During generation, the model is subtly biased to sample more frequently from the "green" list, which acts as a watermark-carrying set, while avoiding tokens in the "red" list. This results in output text that biases outputs toward "green" tokens with minimal quality loss. Variants aim to improve robustness and preserve quality (Liu et al., 2024; Zhao et al., 2023; Hou et al., 2024).

More recently, commercial systems have also entered this space. For example, Google's SynthID-Text watermarking system employs a strategy called Tournament Sampling, in which candidate tokens are ranked according to randomized watermarking functions, and the highest-ranked token is selected during generation (Dathathri et al., 2024). While both academic and commercial watermarking approaches have shown promise, they are primarily evaluated on general-purpose domains such as news or encyclopedic text, and rarely tested under the stylistic and ethical constraints found in peer review.

While a few frameworks target peer review watermarking (Rao et al., 2025), they rely on tightly integrated pipelines and lack evaluation across adaptation modes. Topic-based watermarking (TBW) (Anonymous, 2024), originally proposed for open-domain text, provides a lightweight, semantically guided alternative.

B Discussion

Topic-based watermarking performs particularly well in the peer review setting due to the natural alignment between the subject of a paper and the content of its corresponding review. Unlike more open-ended generation tasks, peer reviews are tightly grounded in the paper being evaluated, making significant topic shifts unlikely, unless introduced deliberately by the reviewer. Since high-quality, relevant reviews are needed for the academic evaluation process, such intentional degradation is improbable in practice.

We also observe that topic-based watermarking is compatible across varying levels of LLM adaptation, from base models to fine-tuned variants. While the few-shot setting shows degradation in detection robustness, we attribute this to topic mismatch between the few-shot exemplars and the review being generated. This limitation can be mitigated with better exemplar selection or dynamic prompt construction.

From a deployment perspective, TBW offers a practical solution for reviewer attribution. The method is efficient and detection incurs minimal computational overhead, making it suitable for integration into existing conference submission pipelines (Anonymous, 2024). Its low latency and lack of architectural modifications make it a compelling candidate for enforcement mechanisms in venues that prohibit LLM-assisted review writing.

Lastly, our evaluation uses a constrained input (title and abstract) due to context window limitations. We expect that access to the full paper would further enhance generation quality and strengthen watermark consistency by grounding outputs in topic-relevant content.

C Peer Review Task Specifics

This appendix provides additional details regarding the peer review generation setup described in §3. Specifically, we include conference-level review statistics and implementation details for TBW as well as prompting and fine-tuning the `Llama-3.1-8B` model.

C.1 Conference Review Statistics

Table 3 reports the number of reviews collected from each ICLR and NeurIPS conference used in our experiments. Only reviews submitted prior to the release of ChatGPT (November 2022) were included to minimize the likelihood of LLM-generated content in the training data. No additional filtering was applied beyond restricting the dataset to pre-ChatGPT conferences where all reviews were used in their original form.

Conference: Year	Number of Reviews
ICLR: 2018	935
ICLR: 2019	1419
ICLR: 2020	2213
ICLR: 2021	2594
ICLR: 2022	2617
ICLR: 2023	3793
NeurIPS: 2021	2768
NeurIPS: 2022	2824

Table 3: Review counts per conference used in training and evaluation. The total number of unique reviews is 19,163.

C.2 TBW Parameter Details

C.2.1 Implementation Components

We use the same core components as the original TBW implementation to ensure consistency. Token and topic embeddings are computed using the `all-MiniLM-L6-v2` sentence embedding model (Reimers and Gurevych, 2020). Topic extraction from input prompts is performed using `KeyBERT` (Grootendorst, 2020) for keyword-based topic identification.

C.2.2 Topic Assignment & Green List Construction

Following the TBW framework, we partition the vocabulary into green lists based on semantic similarity to our predefined set of $K = 4$ domain-specific topics: {`theory`, `applications`, `models`, `optimization`}. Each token $v \in V$ is assigned to the green list G_{t_i} if its cosine similarity with topic t_i exceeds threshold τ:

$$\text{sim}(v, t_i) = \frac{\mathbf{e}_v \cdot \mathbf{e}_{t_i}}{\|\mathbf{e}_v\| \|\mathbf{e}_{t_i}\|}$$

Tokens not meeting the similarity threshold are placed in a residual set and evenly distributed across all green lists to maintain complete vocabulary coverage.

C.2.3 Generation & Detection Parameters

We apply a logit bias of $\delta = 2.0$ to green-list tokens during generation, consistent with values reported in prior watermarking literature (Kirchenbauer et al., 2023). For token-to-topic assignment, we primarily use a cosine similarity threshold of $\tau = 0.7$, with additional evaluation at $\tau = 0.3$ to assess how watermark detection and text quality vary under relaxed alignment constraints.

For detection, we use the statistical z-test with baseline proportion γ set to the expected green-token rate under random sampling, and threshold $z_{\text{threshold}}$ tuned for desired false positive rates.

C.3 Model Configuration Details

C.3.1 Few-shot Prompting Setup

In the few-shot setting, the model is given a prompt containing a paper's title and abstract followed by a fixed instruction:

> **Title:** [TITLE]
> **Abstract:** [ABSTRACT]
> Please write a detailed review.

Each prompt includes two example reviews prepended to help the model learn the expected structure and tone of a review. These few-shot examples are randomly sampled from the training pool but excluded from evaluation generations. Specifically, the two examples prepended to each prompt are drawn from the first two entries in the fine-tuning training split, ensuring consistency across models.

C.3.2 Fine-Tuning Setup

For fine-tuning, we follow a supervised instruction-tuning setup where each instance consists of an input prompt (title + abstract + instruction) and a target completion (review text). The dataset is split into training (80%), validation (10%), and test (10%) subsets. We fine-tune using LoRA (Low-Rank Adaptation) with 4-bit quantization, enabling gradient checkpointing and early stopping. The objective is to improve the fluency and consistency of generated reviews while approximating the tone and structure typical of human-written peer reviews.

For instruction-tuned generation, we fine-tune the `Llama-3.1-8B` model using a parameter-efficient LoRA (Low-Rank Adaptation) method. LoRA freezes the original model weights and injects trainable low-rank matrices into a subset of layers, enabling effective fine-tuning with a small number of additional parameters. This approach is well-suited for large-scale models, reducing memory usage and training time while maintaining performance. Key settings are described in Table 4. All experiments were conducted using the Hugging Face `Transformers` and `PEFT` libraries, with training orchestrated using the `Trainer` API. The final adapters and tokenizer were saved for downstream evaluation. The dataset consists of the prompt (title, abstract, and generation instruction) and a completion (review text), compatible with instruction tuning for causal language models.

Parameter	Value
Adapter type	LoRA
LoRA r/α	16/32
LoRA dropout	0.1
Training epochs	3
Batch size (per device)	2
Max sequence length	2048 tokens
Learning rate	1e-4
Warmup ratio	0.2
Quantization	4-bit (NF4), double quantization
Target modules	`q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`

Table 4: Fine-tuning Hyperparameters

D Generation Quality Evaluations

We expend our evaluation of topic-based watermarking (TBW) to assess its sensitivity to different token-to-topic similarity thresholds. Specifically, we re-run perplexity and BERTScore evaluations using a lower semantic similarity threshold of $\tau = 0.3$ (vs. $\tau = 0.7$ in the main experiments). We also provide BERTScore comparisons of TBW ($\tau = 0.7$) against KGW and SynthID baseline watermarking schemes.

D.1 Baseline Watermarking Parameters

We compare TBW against two established watermarking methods: KGW (Kirchenbauer et al., 2023), a pioneering approach for LLM watermarking, and SynthID-Text (Dathathri et al., 2024), Google's proprietary technique for text attribution. All baseline implementations use the open-source MarkLLM framework (Pan et al., 2024) with the following configurations:

KGW Parameters:

```
{
  "gamma": 0.5,
  "delta": 2.0,
  "prefix_length": 1,
  "z_threshold": 4.0
}
```

SynthID Parameters:

```
{
  "ngram_len": 5,
  "watermark_mode": non-
      distortionary,
  "threshold": 0.52,
  "detector_type": mean
}
```

D.2 Sample Retention Counts

Table 5 shows the number of samples retained with perplexity ≤ 20 for the main experiments using $\tau = 0.7$.

D.3 BERTScore Evaluation (TBW $\tau = 0.7$)

We use BERTScore F1 to evaluate semantic similarity between generated reviews and ground-truth references. This metric compares contextual embeddings and is tolerant to paraphrasing, making it well-suited for open-ended review generation. Results across all model configurations are shown in Figure 2.

TBW causes only a minor drop in BERTScore, indicating that semantic fidelity is largely preserved. Notably, in the base model, TBW narrows

Model	Scheme	Samples Retained
Base	NW	508
	TBW	**991**
	KGW	840
	SynthID	538
Few-shot	NW	**1000**
	TBW	**1000**
	KGW	**1000**
	SynthID	**1000**
Fine-tuned	NW	**1000**
	TBW	**1000**
	KGW	**1000**
	SynthID	**1000**

Table 5: Number of retained generations with perplexity ≤ 20, comparing no watermark (NW) and TBW ($\tau = 0.7$) across model configurations.

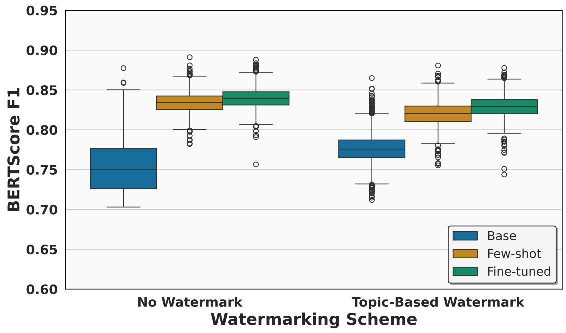


Figure 2: BERTScore F1 distributions across model configurations with and without TBW ($\tau = 0.7$). Higher values indicate greater semantic similarity to the ground truth.

the BERTScore distribution, suggesting more consistent alignment with the source prompt across samples.

D.4 BERTScore Evaluation (KGW & SynthID)

We evaluate BERTScore F1 for generations produced with KGW and SynthID. Results are presented in Figure 3.

In the few-shot and fine-tuned configurations, KGW performs comparably to TBW at $\tau = 0.7$, with similar median BERTScore values and distributional tightness. However, in the base model configuration, KGW shows a broader distribution of scores, indicating higher variability in semantic fidelity. This suggests that KGW, like TBW, is more effective when the generation is guided by conditioning or domain adaptation. SynthID shows a similar pattern but with slightly more pronounced effects. In the base model, SynthID outputs exhibit a wider spread compared to both TBW and KGW, reflecting less stable semantic alignment. In contrast, SynthID performs slightly better in the few-shot and fine-tuned settings, with a 1–2% improvement in BERTScore F1 over TBW at $\tau = 0.7$.

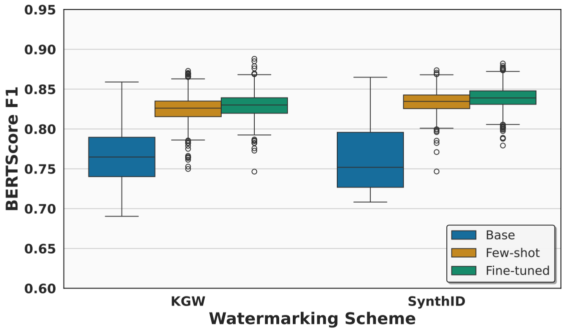


Figure 3: BERTScore F1 distributions across model configurations with KGW and SynthID. Higher values indicate greater semantic similarity to the human-written reference.

These results highlight that while all watermarking methods introduce some tradeoff between attribution and quality, their semantic fidelity is more stable in strongly conditioned generation settings. SynthID offers stronger semantic preservation under tight generation constraints, but at the cost of higher perplexity and fluency degradation in lower-context scenarios.

D.5 Evaluation with Lower Topic Similarity Threshold ($\tau = 0.3$)

We repeat the perplexity and BERTScore evaluations using a relaxed topic assignment threshold of $\tau = 0.3$. This setting allows more tokens to be included in each green list, resulting in stronger watermark signals but potentially greater degradation in generation quality. The results help assess how sensitive TBW is to this design parameter.

D.5.1 Perplexity

Figure 4 shows the perplexity distributions for all model configurations, comparing outputs generated with and without TBW under $\tau = 0.3$. Following the same visualization protocol as in the main paper, we truncate values above 20 for readability. Table 6 reports how many samples remained below this threshold in each setting.

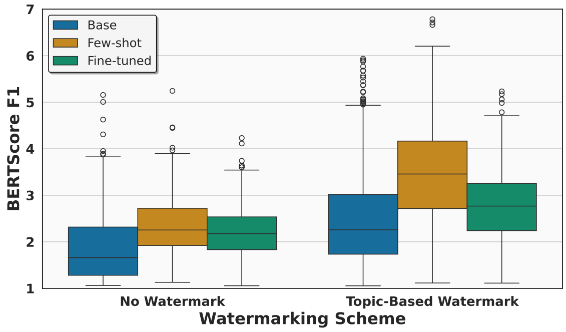


Figure 4: Perplexity distributions across model configurations with and without TBW ($\tau = 0.3$). Lower values indicate better fluency. Values above 20 are truncated for clarity.

Model	Scheme	Samples Retained
Base	NW	508
	TBW	**684**
Few-shot	NW	**1000**
	TBW	**1000**
Fine-tuned	NW	**1000**
	TBW	**1000**

Table 6: Number of generations with perplexity ≤ 20, comparing unwatermarked (NW) and TBW outputs ($\tau = 0.3$).

As expected, TBW at $\tau = 0.3$ produces slightly higher perplexity than unwatermarked generations, reflecting modest fluency degradation. Compared to TBW at $\tau = 0.7$, this lower-threshold variant results in fewer retained samples in the base model (684 vs. 991), suggesting increased fluency loss under weaker semantic filtering. Additionally, there is worse performance in the few-shot model, consistent with less effective topic alignment, but with improved perplexity in the fine-tuned model potentially due to the broader green lists better overlap with the model's learned domain-specific vocabulary.

These results support the view that τ serves as a tradeoff between watermark strength and generation quality, and that optimal settings may vary depending on the model's adaptation level.

D.5.2 BERTScore Evaluation

We repeat the BERTScore F1 evaluation, using generations produced with TBW at $\tau = 0.3$. Results are shown in Figure 5.

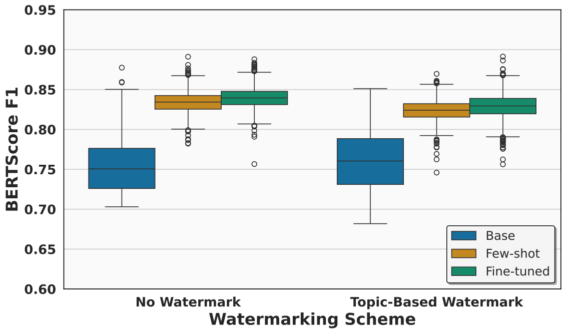


Figure 5: BERTScore F1 distributions across model configurations with and without TBW ($\tau = 0.3$). Higher values indicate greater semantic similarity to the human-written reference.

We observe that TBW with $\tau = 0.3$ results in similar BERTScore degradation as seen with $\tau = 0.7$ in both the few-shot and fine-tuned model configurations. This indicates that semantic fidelity is largely preserved even with a broader green list, suggesting the robustness of TBW's semantic biasing strategy in these more guided generation settings.

However, the base model configuration shows more pronounced differences. Compared to TBW at $\tau = 0.7$, the base model with $\tau = 0.3$ produces generations with a broader range of BERTScore values, indicating increased variability in semantic alignment. This dispersion suggests that, in the absence of stronger conditioning (e.g., few-shot or fine-tuning), relaxing the similarity threshold introduces more topical drift, potentially reducing TBW's ability to maintain consistent semantic guidance.

These results reinforce that TBW is more stable in controlled generation setups, while its performance in lower-context settings (like the base model) is more sensitive to the choice of τ.

E Robustness Evaluations

We provide comprehensive robustness analysis for the evaluations described in §4.2, including detailed performance metrics and ROC curve analysis across all experimental conditions.

Figure 6 presents ROC curves for TBW detection performance across base, few-shot, and fine-tuned model configurations under three conditions: no attack baseline, PEGASUS paraphrasing, and DIPPER paraphrasing. The curves demonstrate

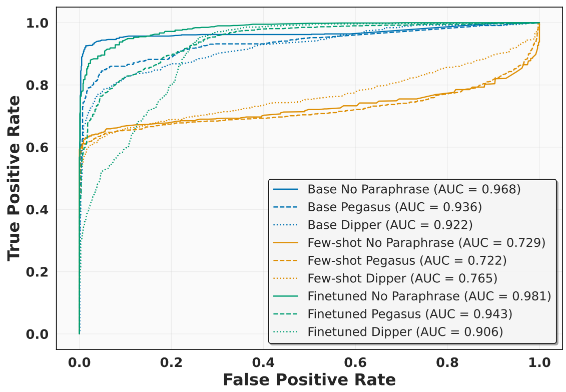


Figure 6: ROC curves for TBW detection under no attack, PEGASUS, and DIPPER paraphrasing, across all model configurations. The curves demonstrate TBW's robustness across attack severity and adaptation settings.

TBW's consistent robustness across varying attack severity and model adaptation settings. Detection performance remains robust in base and fine-tuned configurations, with area under the curve (AUC) values exceeding 0.90 under no attack conditions and experiencing only moderate degradation under paraphrasing attacks. The few-shot model exhibits greater sensitivity to paraphrasing-induced topic dilution, as discussed in §4.2, resulting in reduced detection confidence and lower overall performance across all attack conditions.

Model	Attack	AUC	F1	TPR@1%	TPR@10%
Base	None	0.968	0.955	0.908	0.956
	PEGASUS	0.936	0.893	0.746	0.861
	DIPPER	0.922	0.857	0.669	0.826
Few-shot	None	0.729	0.768	0.626	0.669
	PEGASUS	0.722	0.758	0.609	0.655
	DIPPER	0.765	0.754	0.565	0.659
Fine-tuned	None	0.981	0.927	0.817	0.948
	PEGASUS	0.944	0.858	0.593	0.826
	DIPPER	0.906	0.861	0.348	0.598

Table 7: Detection performance across model configurations and attack settings. Metrics include ROC-AUC, best F1 score, and true positive rate (TPR) at fixed false positive rates (FPRs) of 1% and 10%.

Table 7 provides comprehensive detection performance across all experimental conditions, including ROC-AUC, F1 scores, and true positive rates (TPR) at relevant false positive rate (FPR) thresholds of 1% and 10%. The base model demonstrates robustness, maintaining an AUC above 0.92 even under the strongest paraphrasing attacks, with F1 scores of 0.893 (PEGASUS) and 0.857 (DIPPER). At the 1% FPR threshold, the base model achieves TPR values of 74.6% and 66.9% under PEGASUS and DIPPER attacks, respectively, indicating strong practical utility for high-precision detection scenarios.

The fine-tuned model shows similar resilience with AUC values of 0.944 (PEGASUS) and 0.906 (DIPPER), though it exhibits greater sensitivity at low FPR thresholds, particularly under DIPPER attacks where TPR@1% drops to 34.8%. This suggests that fine-tuning may increase vulnerability to certain paraphrasing patterns while maintaining overall detection capability.

In contrast, the few-shot configuration shows limited degradation under attacks, with AUC values remaining stable around 0.72-0.77 across all conditions. However, the consistently lower baseline performance (AUC = 0.729) indicates that topic misalignment limits detectability in this setting, making the relative robustness less operationally significant.

The maintained performance at low FPR thresholds across model configurations confirms that TBW's vocabulary partitioning strategy effectively preserves detection capability while minimizing false alarms on human-written content, as evidenced by the consistent TPR@1% and TPR@10% metrics across experimental conditions.

F Classifier Specifics

We provide implementation details for the classification experiments described in § 4.3. We outline the training setup used for both BERT and RoBERTa classifiers and summarize the evaluation strategy for attribution analysis on generated peer reviews.

F.1 Data Construction & Protocol

We first construct a labeled dataset by extracting review texts from our generation pipeline and assigning a class label based on the associated ground truth rating (e.g., scores 1–4 mapped to `reject`, 5–6 to `borderline`, and 7–10 to `accept`). To ensure accurate mapping, we align generated reviews with their original metadata using paper titles as unique identifiers. The final dataset consists of generated reviews paired with class labels, drawn from the fine-tuned generation split described in § C.3.2.

The final dataset consists of generated reviews paired with class labels, drawn from our experimental pipeline. The dataset is stratified into training and held-out test splits, with 9,000 balanced training samples (3,000 per class) and 1,000 test samples for evaluation.

F.2 Classifier Training

For reproducibility, we provide the specific training parameters used to fine-tune our LLM classifiers for predicting peer review labels corresponding to paper rating categories: `reject`, `borderline`, and `accept`.

Each model is fine-tuned using the Hugging Face `Trainer` API with early stopping based on F1. Key training settings described in Table 8. Tokenization was performed using each model's pretrained tokenizer. A padding-aware data collator was used for batch construction. All training was conducted using the Hugging Face `Transformers` library and saved checkpoints were used for downstream evaluation on generated samples.

Parameter	Value
Model types	`bert-base-uncased`, `roberta-large`
Number of classes	3 (`reject`, `borderline`, `accept`)
Max sequence length	512 tokens
Training epochs	5
Batch size (per device)	16
Learning rate	2e-5
Warmup ratio	0.1
Optimizer	AdamW
Scheduler	Cosine with restarts
Dropout	0.2 (attention and hidden layers)
Gradient clipping	Max norm 1.0
Label smoothing	0.1
Precision	Mixed (FP16 with full-eval)
Quantization	4-bit weight loading
Evaluation strategy	Per epoch; best model selected via F1
Early stopping	Enabled (patience = 1)

Table 8: Classifier Training Hyperparameters

F.3 Classifier Evaluation

We evaluate both BERT and RoBERTa classifiers on a held-out test set of 1,000 human-written peer reviews. This evaluation step assesses whether the models can correctly recover the original review rating category (`reject`, `borderline`, `accept`) before applying them to generated or watermarked samples.

Predictions are obtained from each trained classifier on the tokenized test set and compared against the ground truth labels. We compute confusion matrices to visualize class-specific misclassification patterns and report overall accuracy as a coarse measure of performance. BERT achieves an accuracy of 51.3%, while RoBERTa performs slightly better at 53.9%. Figures 7 and 8 present the confusion matrices for BERT and RoBERTa, respectively.

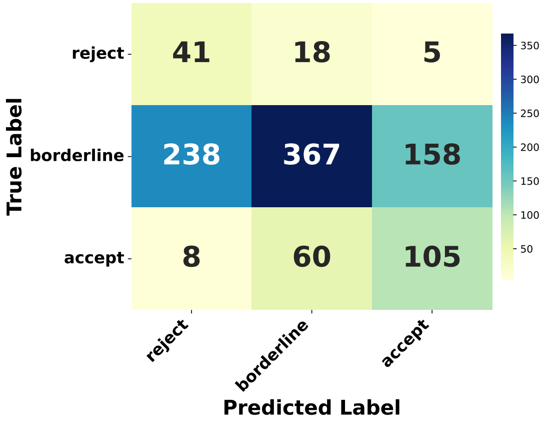


Figure 7: Confusion matrix for the BERT classifier on 1,000 human-written peer reviews.

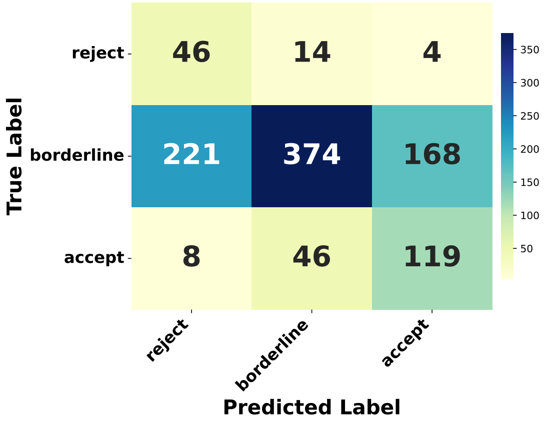


Figure 8: Confusion matrix for the RoBERTa classifier on on 1,000 human-written peer reviews.

Both classifiers exhibit a strong predictive tendency toward the `borderline` class. As shown in the confusion matrices, the majority of `borderline` samples are correctly classified by both BERT (367/763) and RoBERTa (374/763). However, a large number of `reject` and `accept` samples are also misclassified as `borderline`. For instance, BERT misclassifies 18 `reject` and 60 `accept` samples as `borderline`, while RoBERTa reduces this to 14 and 46, respectively. Compared to BERT, RoBERTa shows slightly improved separation between all three classes, with fewer misclassifications across off-diagonal entries. In particular, it shows higher retention of true `reject` and

Table 9: Classification performance for topic-based watermarking (TBW) at a lower similarity threshold of $\tau = 0.3$. Results are shown across all model configurations (base, few-shot, fine-tuned) and for both BERT and RoBERTa classifiers.

Classifier	Model	Accuracy	Precision	Recall	F1
BERT	Base	0.289	0.322	0.322	0.288
	Few-shot	0.387	0.334	0.342	0.333
	Fine-tuned	0.414	0.372	0.366	0.360
RoBERTa	Base	0.438	0.338	0.340	0.332
	Few-shot	0.360	0.339	0.344	0.335
	Fine-tuned	0.398	0.375	0.368	0.361

accept labels, suggesting better overall discriminative performance.

F.4 Class-Specific Classifier Evaluation

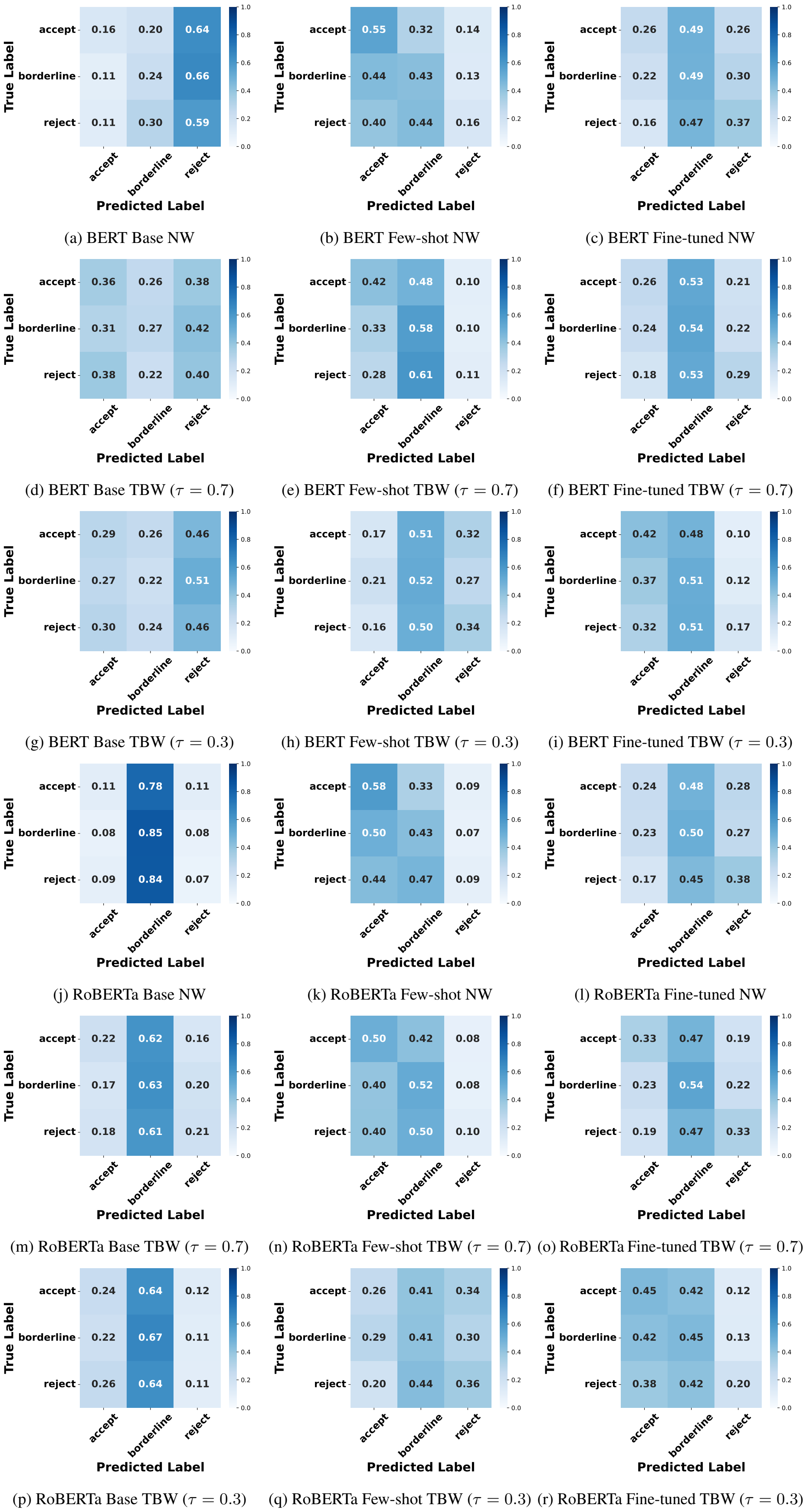
To further characterize classifier performance, we conduct a comprehensive class-specific evaluation of generated peer reviews based on the classification framework introduced in §4.3. This analysis extends the aggregate metrics reported in Table 2 by examining model behavior across the three target rating categories under different watermarking conditions and topic similarity thresholds.

We examine confusion matrices for each classifier (BERT and RoBERTa), stratified by language model configuration (base, few-shot, fine-tuned) and watermarking condition. Additionally, we extend our analysis to topic-based watermarking (TBW) applied at a lower semantic similarity threshold of $\tau = 0.3$, which relaxes the token-to-topic alignment constraints, thereby increasing green-list coverage and watermark signal strength while potentially impacting semantic coherence. Figure 9 presents the complete set of confusion matrices across all configurations, including both $\tau = 0.7$ and $\tau = 0.3$ conditions.

BERT exhibits distinct patterns across model configurations and watermarking conditions. In the base non-watermarked condition (panel a), BERT shows a skew toward the reject column, while the few-shot variant (panel b) demonstrates higher predictions in the accept and borderline columns. The fine-tuned non-watermarked model (panel c) shows the highest concentration in the borderline column, though values remain below 0.50. Under watermarking conditions, BERT base models with both TBW $\tau = 0.7$ and $\tau = 0.3$ (panels d, g) exhibit a slight skew toward the reject column but with modest values barely exceeding 0.50. For few-shot and fine-tuned watermarked variants (panels e, f, h, i), predictions concentrate in the borderline column, with accept predictions consistently higher than reject but lower than borderline.

RoBERTa demonstrates more consistent patterns with clearer biases toward specific categories. Across all base configurations, non-watermarked, TBW $\tau = 0.7$, and TBW $\tau = 0.3$ (panels j, m, p), there is a pronounced bias toward the borderline column. The few-shot and fine-tuned variants generally show better-balanced distributions with higher concentrations in both accept and borderline columns. Notable exceptions include the RoBERTa few-shot TBW $\tau = 0.3$ condition (panel q), which maintains high borderline predictions, and the fine-tuned TBW $\tau = 0.7$ variant (panel o), which also shows elevated borderline concentrations.

Table 9 reports the classification metrics for each classifier and LLM model variant under TBW with $\tau = 0.3$. While overall performance remains comparable to the $\tau = 0.7$ condition, we observe that the fine-tuned model achieves the highest accuracy across both BERT and RoBERTa classifiers, suggesting that domain adaptation remains a dominant factor in attribution effectiveness even under relaxed topic alignment. This analysis underscores the relative semantic distinctiveness of strongly positive (accept) and moderate (borderline) reviews, while highlighting the challenges involved in distinguishing lower-quality (reject) reviews, which often exhibit more linguistic and structural variability across different watermarking configurations and similarity thresholds.



(a) BERT Base NW

(b) BERT Few-shot NW

(c) BERT Fine-tuned NW

(d) BERT Base TBW ($\tau = 0.7$)

(e) BERT Few-shot TBW ($\tau = 0.7$)

(f) BERT Fine-tuned TBW ($\tau = 0.7$)

(g) BERT Base TBW ($\tau = 0.3$)

(h) BERT Few-shot TBW ($\tau = 0.3$)

(i) BERT Fine-tuned TBW ($\tau = 0.3$)

(j) RoBERTa Base NW

(k) RoBERTa Few-shot NW

(l) RoBERTa Fine-tuned NW

(m) RoBERTa Base TBW ($\tau = 0.7$)

(n) RoBERTa Few-shot TBW ($\tau = 0.7$)

(o) RoBERTa Fine-tuned TBW ($\tau = 0.7$)

(p) RoBERTa Base TBW ($\tau = 0.3$)

(q) RoBERTa Few-shot TBW ($\tau = 0.3$)

(r) RoBERTa Fine-tuned TBW ($\tau = 0.3$)

Figure 9: Comparison of confusion matrices across different model configurations and watermarking settings. Top row shows BERT results, middle row shows RoBERTa results, with columns representing no watermarking (NW), topic-based watermarking at $\tau = 0.7$, and topic-based watermarking at $\tau = 0.3$.

Classifier	Model	Watermark	PEGASUS Shifts	DIPPER Shifts
BERT	Base	No watermark	58	54
		TBW ($\tau = 0.7$)	37	23
		TBW ($\tau = 0.3$)	51	45
	Few-shot	No watermark	24	14
		TBW ($\tau = 0.7$)	24	24
		TBW ($\tau = 0.3$)	24	22
	Fine-tuned	No watermark	27	20
		TBW ($\tau = 0.7$)	15	15
		TBW ($\tau = 0.3$)	25	15
RoBERTa	Base	No watermark	13	9
		TBW ($\tau = 0.7$)	23	25
		TBW ($\tau = 0.3$)	16	19
	Few-shot	No watermark	30	13
		TBW ($\tau = 0.7$)	27	22
		TBW ($\tau = 0.3$)	25	20
	Fine-tuned	No watermark	24	14
		TBW ($\tau = 0.7$)	18	22
		TBW ($\tau = 0.3$)	21	18

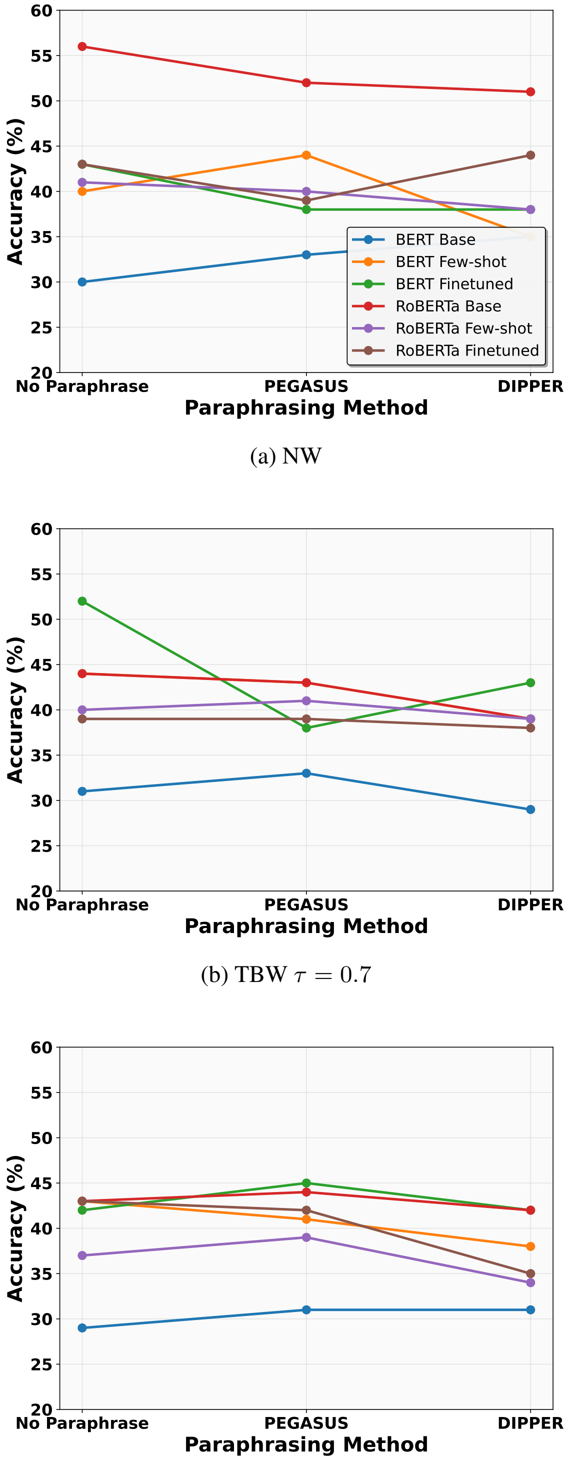
Table 10: Number of review classification shifts under paraphrasing attacks. Each entry reflects the count (out of 100 paraphrased samples) where the predicted class label differs from the original. Results are grouped by classifier, model variant, and watermarking scheme (NW, TBW ($\tau = 0.7$), TBW ($\tau = 0.3$)), and evaluated separately under PEGASUS and DIPPER paraphrasing models.

F.5 Peer Review Shifts Under Paraphrasing

To evaluate the impact of paraphrasing on classifier-based review attribution, we examine both classification accuracy and label stability under two paraphrasing threat models: PEGASUS and DIPPER. Specifically, we sample 100 LLM-generated peer reviews and apply paraphrasing to each using both models. We then assess the classification performance before and after paraphrasing under three watermarking conditions: no watermark (NW), topic-based watermarking (TBW) with $\tau = 0.7$, and TBW with $\tau = 0.3$.

Figure 10 presents accuracy changes across all classifier and model configurations. Table 10 reports the number of label transitions (e.g., `Accept` → `Borderline`) observed in the paraphrased reviews. These metrics reflect the semantic resilience of reviewer intent and classification stability under adversarial rewording.

Our results indicate that paraphrasing generally reduces classification accuracy across all settings, though the degree of degradation varies. Notably, TBW models exhibit consistent accuracy declines under paraphrasing for both τ values, suggesting that watermarked outputs are more sensitive to adversarial modification in terms of downstream attribution. In contrast, non-watermarked outputs show mixed effects while some configurations experience accuracy drops, others see minor improvements. We attribute this to incidental lexical clarifications introduced by the paraphrasers. In terms of label stability, TBW reduces the number of class shifts compared to the non-watermarked baseline. This trend is especially evident under the PEGASUS paraphrasing model, where non-watermarked outputs exhibit the highest number of shifts. These findings suggest that TBW not only leaves a detectable signature but may also provide a degree of structural regularity that preserves classification under text manipulation.



(a) NW

(b) TBW $\tau = 0.7$

(c) TBW $\tau = 0.3$

Figure 10: Classification accuracy on paraphrased peer reviews across three watermarking settings: (a) no watermark (NW), (b) topic-based watermarking (TBW) with $\tau = 0.7$, and (c) TBW with $\tau = 0.3$. Results are shown across all model configurations (base, few-shot, fine-tuned) for both BERT and RoBERTa classifiers under PEGASUS and DIPPER paraphrasing attacks.