

The Feasibility of Topic-Based Watermarking on Academic Peer Reviews

Alexander Nemecek, Yuzhou Jiang, Erman Ayday

Case Western Reserve University
{aj98, yxj466, exa208}@case.edu

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 lightweight, semantic-aware technique designed to embed detectable signals into LLM-generated text. We conduct a comprehensive 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 strong robustness to paraphrasing-based evasion. These findings highlight the viability of TBW as a minimally intrusive and practical solution for enforcing LLM usage in peer review.

1 Introduction

As large language models (LLMs) continue to evolve, their adoption has accelerated, particularly in academic writing (Dergaa et al., 2023; Editorials, 2023). LLMs are widely used for language polishing, literature search, and low-novelty writing, often producing text nearly indistinguishable from human-authored content. Many conferences now explicitly permit authors to use LLMs for low-novelty 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 to third-party systems (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 findings underscore the urgency of developing attribution mechanisms to detect and manage unauthorized LLM usage.

As LLM-generated content increasingly mirrors human writing, distinguishing between machine- and human-authored reviews has become difficult. Stylistic cues alone are insufficient for reliable attribution, especially in the absence of disclosure (Mitchell et al., 2023). This creates an urgent need for technical mechanisms to trace the provenance of peer reviews. A widely explored approach is *watermarking*, which has been adopted across various domains to embed imperceptible, machine-detectable signatures into generated text (Zhao et al., 2024). Recent methods, such as topic-based watermarking, bias generations toward semantically aligned tokens that are robust and minimally intrusive. However, existing work focuses on general-domain text, with limited analysis in peer review (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. Rather than proposing a new algorithm, we apply an existing lightweight, topic-guided watermarking scheme to this domain-specific, policy-sensitive task. Topic-based watermarking (TBW) offers a balance of efficiency, robustness to paraphrasing, and minimal impact on generation quality, making it suitable for peer re-

view, where stylistic fidelity and semantic coherence are critical. It also supports domain adaptation through customizable topic lists, aligning well with the structured topical nature of peer reviews. 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 reliable 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 quality, semantic preservation, classifier-based attribution, and robustness to paraphrasing. We further compare TBW against general-purpose watermarking methods 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.

2 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 watermark-

ing 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) (Nemecsek et al., 2024), originally proposed for open-domain text, provides a lightweight, semantically guided alternative. We adapt TBW to peer review by aligning token selection with domain-relevant topics, preserving generation quality while supporting practical reviewer attribution. Section 3.2 details this adaptation.

3 Methodology

Our goal is to evaluate the applicability of topic-based watermarking in the domain of academic peer review. We investigate whether such watermarking can preserve the quality and semantic integrity of generated reviews, while enabling robust attribution under paraphrasing attacks. We describe our data collection, model configurations, watermarking integration, and evaluation procedures.

3.1 Peer Review Generation Task

We simulate realistic LLM-based peer review generation by training and prompting language models to write reviews conditioned on a paper’s title and abstract. We use the abstract rather than the full paper because full submissions often exceed typical context window limits and are less readily available in structured form. This section describes the dataset used, the model variants we examine, and our prompting and fine-tuning strategies.

3.1.1 Dataset

To evaluate topic-based watermarking in the context of peer review, we compile a dataset of paper titles, abstracts, and corresponding reviews from ICLR and NeurIPS conferences using the OpenReview API (OpenReview, 2024). Each review includes a summary, strengths and weaknesses, and a final recommendation score. To minimize the risk of including LLM-generated reviews, we restrict our dataset to conferences held before the public release of ChatGPT (November 2022) (OpenAI, 2022). Specifically, we collect reviews from ICLR 2018–2023 and NeurIPS 2021–2022, noting that the ICLR 2023 review phase, despite the

conference date, occurred prior to ChatGPT’s availability (ICLR, 2023). Although language models existed before this, they were not widely adopted in peer review workflows at scale.

The final dataset contains approximately 19,000 reviews. For each paper, we randomly sample a single review to construct prompt-completion training pairs, ensuring diversity in reviewer perspectives while avoiding overrepresentation of any one submission. Detailed review counts by conference are provided in Appendix A.1.

3.1.2 Model Configurations

To assess the feasibility of topic-based watermarking across varying levels of model adaptation and reviewer effort, we utilize the Llama-3.1-8B (Grattafiori et al., 2024) open-source language model in three configurations: base, few-shot, and fine-tuned. The base configuration uses the pretrained model without any 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 better replicate the expected format and tone with lightweight guidance. Finally, the fine-tuned configuration involves additional supervised training on peer review data using parameter-efficient methods, resulting in a model that is more aligned with the review-writing task and capable of generating coherent, domain-adapted outputs. This model size offers a practical balance between computational efficiency and generation quality, making it suitable for experiments involving multiple training configurations.

3.1.3 Prompting and Few-shot Learning

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.

3.1.4 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. Fine-tuning hyperparameters, model setup, and training procedure details are provided in Appendix A.2.

3.2 Topic-Based Watermarking

Topic-based watermarking (TBW) (Nemecek et al., 2024) is a semantic-aware watermarking method 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. This design helps preserve fluency and coherence while enhancing robustness against paraphrasing and token-level edits. We briefly summarize the TBW generation and detection process as applied in our setup.

3.2.1 Token-to-Topic Mappings

TBW first assigns tokens to topic-specific green lists using semantic similarity. A small set of generalized topics t_1, \dots, 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 calculated:

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

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.

While the original implementation used general-purpose topic categories (e.g., technology,

sports), we adapt the topic set to align with the thematic structure of academic reviews, better reflecting the linguistic and topical distribution of this domain.

3.2.2 Generation

Once topic-specific green lists are defined, TBW applies a watermark by biasing the model’s output distribution during generation. For each input prompt, the most relevant topic is identified using a lightweight keyword extraction method (e.g., KeyBERT). If the extracted topic exactly matches one of the predefined topic labels, the corresponding “green” list is selected. If no exact match is found, topic embeddings are computed and the most similar predefined topic is selected based on cosine similarity.

At each decoding step, the model produces a probability distribution over its vocabulary V . TBW modifies this distribution by adding a small logit bias δ to all tokens in the selected green list. This increases the likelihood of sampling topic-aligned tokens after applying the softmax function, subtly guiding the generation process without altering the model architecture or requiring multiple decoding passes. The watermark strength is controlled by the value of δ : higher values produce stronger attribution signals but cause detectable shifts in word choice or token distribution. The approach is model-agnostic and incurs minimal overhead, making it compatible with standard generation pipelines.

3.2.3 Detection

TBW uses a statistical test to detect whether a given text contains a watermark. Detection mirrors the generation process by recovering the relevant topic inferred from the input text using the same keyword or embedding-based matching procedure, and the corresponding green list G_{t^*} is recovered.

The number of green-list tokens g is then counted in the text \mathbf{z}_{test} , and compared to the total number of 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. The threshold can be tuned to balance sensitivity and specificity, and the method is prompt-

and model-agnostic at inference time, requiring access only to the generated output. Importantly, the detection process is model-agnostic and does not require access to the model logits or original input prompt.

3.2.4 Watermarking Configurations

To ensure consistency with the original TBW implementation while adapting it to the domain of peer review, we retain most of the original parameter settings. We use the same sentence embedding model, all-MiniLM-L6-v2 (Reimers and Gurevych, 2020), to encode tokens and topic labels into a shared semantic space. Topic extraction from input prompts is performed using KeyBERT (Groendorst, 2020), as in the original work.

Following the TBW framework, we partition the vocabulary into green lists based on semantic similarity to a predefined set of $K = 4$ topics. While the original implementation used general-purpose topics such as {animals, technology, sports, medicine}, we adapt these categories to reflect the structure and content of machine learning conference reviews. Specifically, we define the following domain-specific topics: {theory, applications, models, optimization}. These topics are designed to capture broad themes in peer review content from venues like ICLR and NeurIPS, and can be adjusted to suit different research domains.

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

3.3 Rationale for Topic-Based Watermarking

While several general-purpose watermarking methods exist, we select topic-based watermarking (TBW) for its unique combination of robustness, adaptability, and minimal performance overhead. Prior work has demonstrated that TBW is resilient to paraphrasing, while preserving generation quality and incurring no additional inference cost (Nemecek et al., 2024). This property is important in the peer review setting, where paraphrasing represents a realistic threat model where a reviewer seeking to obscure LLM use may rewrite or rephrase parts of a generated review, but is unlikely to intro-

duce noise or degrade the review’s usefulness or semantic integrity. These full-paraphrase attacks, rather than token-level perturbations or synthetic distortions, reflect plausible reviewer behavior under current policy constraints.

TBW’s semantic token-level biasing strategy is well-suited to this context. It subtly steers generation toward topic-consistent vocabulary without disrupting fluency or style, both of which are critical in high-stakes peer review writing. In addition, TBW supports domain adaptation through customizable topic lists, and relies on a topic-matching assumption that naturally holds in peer review, where content is expected to stay aligned with the paper under evaluation.

Finally, the peer review task inherently satisfies TBW’s core assumption of topic consistency between the prompt and the generated output. One known limitation of TBW is the *Topic Matching Assumption*, which requires that the generated text remain semantically aligned with the prompt topic. In general-purpose settings, this assumption can be violated due to topic drift or open-ended generation. In peer review, however, this risk is minimal, as the input (e.g., paper title and abstract) directly constrains the review content. A reviewer cannot reasonably produce a review on a different topic than the paper itself. As such, TBW aligns naturally with the structural and semantic constraints of the peer review task.

4 Experiments

To evaluate the applicability of topic-based watermarking (TBW) in the domain of peer review, 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 the impact of TBW on peer review generation, we evaluate outputs using perplexity and BERTScore (Zhang et al., 2019). Following prior work (Nemecek et al., 2024), we apply a semantic similarity threshold of $\tau = 0.7$ to construct topic-aligned green lists. We use 1,000 samples per model configuration (base, few-shot, fine-tuned), each consisting of approximately 200 ± 5 tokens. Additional results for a lower threshold ($\tau = 0.3$) and comparisons to baseline watermarking schemes are provided in Appendix B.

4.1.1 Perplexity

We compute perplexity using the same model that generated the text (Llama-3.1-8B), with lower values indicating higher fluency. Values above 20 are truncated in visualizations for readability (Figure 1), and the number of retained samples is shown in Table 1. This setup reflects how confidently the model assigns probability to its own output, serving as a proxy for fluency.

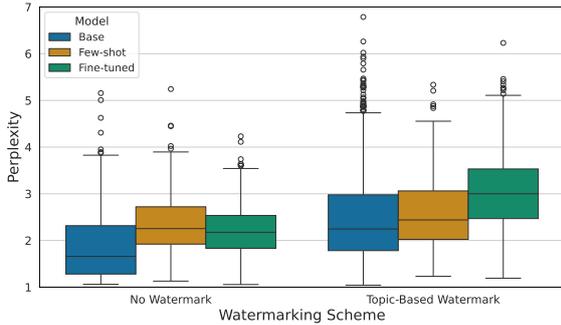


Figure 1: Perplexity distributions across model configurations with and without TBW ($\tau = 0.7$). Lower values indicate better fluency. Values above 20 are truncated for clarity.

Model	Scheme	Samples Retained
Base	NW	508
	TBW	991
Few-shot	NW	1000
	TBW	1000
Fine-tuned	NW	1000
	TBW	1000

Table 1: Number of retained generations with perplexity ≤ 20 , comparing no watermark (NW) and TBW across model configurations.

TBW introduces only a slight increase in perplexity, consistent with prior findings (Nemecek et al., 2024). In the base model, over 50% of unwatermarked generations exceed a perplexity of 20, while nearly all TBW outputs fall below this threshold. This suggests that TBW preserves naturalness and may even enhance lexical consistency in low-context settings by nudging generation toward topic-relevant vocabulary.

4.1.2 BERTScore Evaluation

We use BERTScore F1 to evaluate semantic similarity between generated reviews and ground-truth

references. This metric, which compares contextual embeddings, is tolerant to paraphrasing and thus well-suited for open-ended review generation. Results across all model configurations are shown in Figure 2.

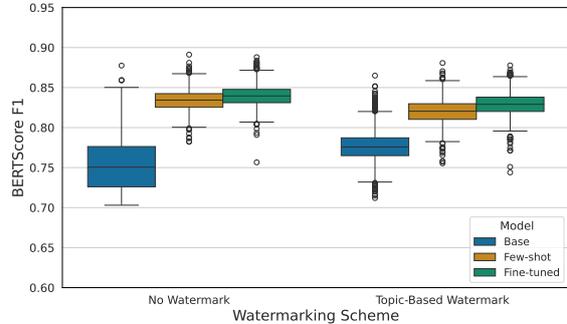


Figure 2: BERTScore F1 distributions across model configurations with and without TBW ($\tau = 0.7$). Higher values indicate greater semantic similarity to the ground truth.

TBW causes only a minor drop in BERTScore, indicating that semantic fidelity is largely preserved. Notably, in the base model, TBW narrows the BERTScore distribution, suggesting more consistent alignment with the source prompt across samples.

4.2 Robustness to Paraphrasing Attacks

We assess TBW’s resilience to paraphrasing attacks, a realistic threat model wherein reviewers may rephrase LLM-generated reviews to evade detection while preserving meaning. We focus on full-paraphrase attacks, which best reflect plausible reviewer behavior, and exclude token-level or partial edits.

To align with prior experiments, we generate 1,000 samples per model (base, few-shot, fine-tuned), each with ~ 200 tokens, using $\tau = 0.7$ for topic alignment. Paraphrasing is applied using PEGASUS and DIPPER, the latter configured with lexical = 60 and order = 40, following standard robustness benchmarks (Hou et al., 2024; Liu and Bu, 2024).

Detection uses the TBW statistical test (see Section 3.2.3), applied to both original and paraphrased generations. Table 2 reports accuracy under three conditions: no paraphrasing, PEGASUS, and DIPPER.

TBW maintains strong robustness in base and fine-tuned models across attack types. The few-

Model	Attack Setting	ROC-AUC	Best F1 Score	TPR@1%FPR	TPR@10%FPR
Base	No Attack	0.9678	0.9546	0.9080	0.9560
	PEGASUS	0.9359	0.8928	0.7460	0.8610
	DIPPER	0.9221	0.8568	0.6690	0.8260
Few-shot	No Attack	0.7286	0.7677	0.6260	0.6690
	PEGASUS	0.7221	0.7584	0.6090	0.6550
	DIPPER	0.7647	0.7537	0.5650	0.6590
Fine-tuned	No Attack	0.9813	0.9266	0.8170	0.9480
	PEGASUS	0.9435	0.8584	0.5930	0.8260
	DIPPER	0.9064	0.8605	0.3480	0.5980

Table 2: 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%.

shot configuration, however, shows reduced recall (0.6260 \rightarrow 0.5650 under DIPPER), likely due to topic mismatch between prompt examples and the target paper, which weakens topic alignment and reduces detectability post-paraphrasing.

Finally, we verify TBW does not yield false positives on human-written reviews, owing to a partitioning strategy that preserves vocabulary diversity across green lists. For full ROC curves and comparisons with baseline watermarking schemes, see Appendix C.

4.3 Classifier-Based Attribution

To complement watermark detection, we evaluate whether LLM-generated peer reviews can be attributed to their original review labels (e.g., accept, borderline, reject) using standard classification models. This task provides a content-based signal of semantic alignment, helping assess whether watermarking affects the interpretability or label consistency of generated reviews. We frame this as a three-way classification problem based on the review score originally assigned to each paper.

4.3.1 Data and Training 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 Section 3.1.4.

We train two transformer-based classifiers,

BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019), to predict the rating category of each review. 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. Tokenization is performed using each model’s native tokenizer, and models are fine-tuned using the HuggingFace Trainer API with early stopping based on F1. We adopt 4-bit precision, label smoothing (0.1), and a cosine learning rate schedule with warmup. Additional training hyperparameters and evaluation on the testing set are provided in Appendix D.

4.3.2 Evaluation

Once trained, both classifiers are applied to a held-out set of generated reviews produced by various generation configurations (base, few-shot, fine-tuned) with and without TBW using $\tau = 0.7$. For each review, we extract the title from the input prompt, retrieve the associated ground truth score from metadata, and map it to a label for evaluation. We evaluate model performance with and without TBW to assess whether watermarking impairs label recoverability.

As shown in Table 3, we observe no degradation in classification performance due to TBW. On the contrary, in most configurations, applying TBW leads to modest improvements in both accuracy and F1. This suggests that topic-based watermarking preserves the semantic structure necessary for accurate label prediction and even enhance it by encouraging more topically consistent language. These findings reinforce TBW’s suitability for attribution tasks in domain-sensitive contexts like peer review, where both traceability and semantic fidelity are critical. Additional analysis on class-

Table 3: Overall classification performance on original LLM-generated reviews. Metrics are averaged over Accept, Borderline, and Reject classes.

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

specifics for human-written reviews is provided in Appendix E and classifier attribution performance under a lower topic similarity threshold ($\tau = 0.3$) to assess the impact of weaker topic alignment is provided in Appendix F. For an analysis of how review content and structure shift under paraphrasing, see Appendix G, which provides changes in accuracy under paraphrasing.

5 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 (Nemecek et al., 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.

6 Conclusion

We present a comprehensive evaluation of topic-based watermarking in the context of academic peer review, a high-stakes domain where LLM use is often restricted but difficult to detect. Unlike prior work that focuses on general-purpose text, our study demonstrates that topic-based watermarking can preserve generation quality, maintain robustness under paraphrasing, and support attribution across different LLM configurations. Its semantic grounding and low computational overhead make it a practical solution for enforcing LLM usage policies in peer review, offering a minimally intrusive mechanism to help safeguard the integrity of academic evaluation.

Limitations

This work inherits a key limitation of topic-based watermarking: the topic-matching assumption. As noted in the original proposal (Nemecek et al., 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 de-

ployed 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.
- 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.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.
- 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.
- ICLR. 2023. Iclr 2023 dates. <https://iclr.cc/Conferences/2023/Dates>. Accessed: May 16, 2025.
- ICML. 2025a. Icm1 2025 call for papers. <https://icml.cc/Conferences/2025/CallForPapers>. Accessed: 2025-05-15.

- ICML. 2025b. Icm1 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.
- 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, Hao-tian 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*.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.
- 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.
- Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, pages 24950–24962. PMLR.
- Alexander Nemecek, Yuzhou Jiang, and Erman Ayday. 2024. Topic-based watermarks for llm-generated text. *arXiv preprint arXiv:2404.02138*.
- 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, Jiao 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*.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.
- 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 Tramèr, 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 Peer Review Task Specifics

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

A.1 Conference Review Statistics

Table 4 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 4: Review counts per conference used in training and evaluation. The total number of unique reviews is 19,163.

A.2 Fine-tuning Details

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 include:

- **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 enabled
- **Target modules:** q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj (These target modules correspond to the attention and MLP projections in transformer layers, where LoRA adapters are most effective.)

All experiments were run 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.

B Generation Quality Evaluations

We expand our evaluation of topic-based watermarking (TBW) to assess its sensitivity to different token-to-topic similarity thresholds. In particular, 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 compare TBW against two baseline watermarking schemes, KGW and SynthID, to contextualize performance. We utilize an open-source watermarking framework, MarkLLM (Pan et al., 2024), and the specified configurations for the baseline watermarking implementations.

B.1 Evaluation with Lower Topic Similarity Threshold ($\tau = 0.3$)

We repeat the perplexity and BERTScore evaluations described in Section 4.1.1 and Section 4.1.2 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.

B.1.1 Perplexity

Figure 3 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 5 reports how many samples remained below this threshold in each setting.

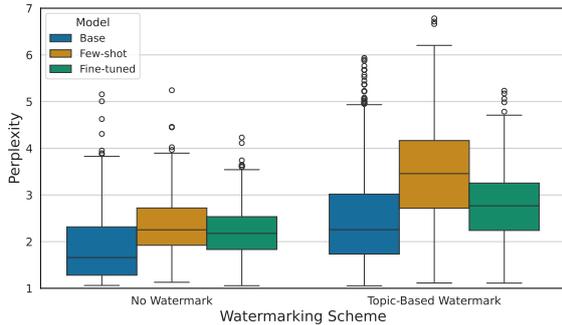


Figure 3: Perplexity distributions across model configurations with and without TBW ($\tau = 0.3$). Lower values indicate better fluency. Values above 20 are truncated for clarity.

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

Table 5: 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.

B.1.2 BERTScore Evaluation

We repeat the BERTScore F1 evaluation under the same setup described in Section 4.1.2, using generations produced with TBW at $\tau = 0.3$. Results are shown in Figure 4.

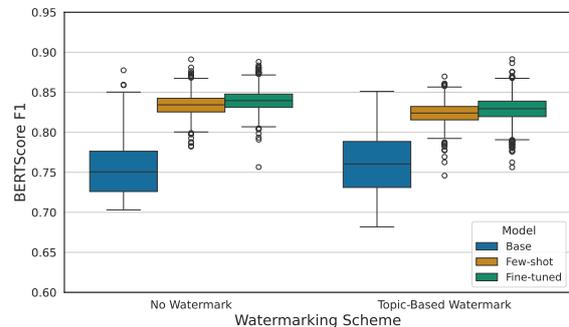


Figure 4: BERTScore F1 distributions across model configurations with and without TBW ($\tau = 0.3$). Higher values indicate greater semantic similarity to the human-written reference.

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 per-

formance in lower-context settings (like the base model) is more sensitive to the choice of τ .

B.2 Baseline Watermarking Quality

We compare TBW against two existing watermarking methods:

- KGW (Kirchenbauer et al., 2023): one of, if not the first watermarking approach for LLMs.
- SynthID-Text (SynthID) (Dathathri et al., 2024): Google’s proprietary watermarking technique designed for text attribution.

We evaluate their impact on fluency using perplexity and semantic similarity using BERTScore.

B.2.1 Perplexity

We evaluate perplexity for generations produced using KGW and SynthID, comparing their impact on fluency using the same evaluation framework as in Section 4.1.1. Figure 5 shows the perplexity distributions for each baseline, while Table 6 reports the number of samples with perplexity ≤ 20 after truncation.

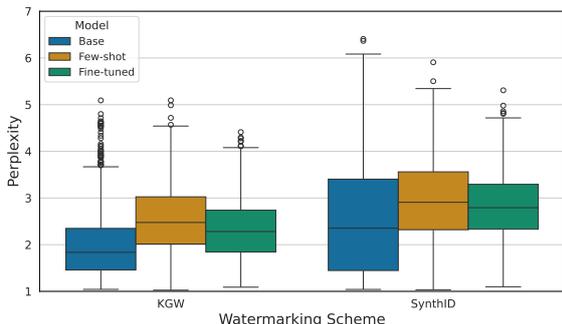


Figure 5: Perplexity distributions across model configurations with KGW and SynthID. Lower values indicate better fluency. Values above 20 are truncated for clarity.

Across all models, KGW performs reasonably well in preserving fluency. In the base model, its perplexity distribution is narrower and more favorable than that of SynthID, with 840 out of 1000 samples retained. In the few-shot setting, KGW is comparable to TBW at $\tau = 0.7$, exhibiting slightly less variability. In the fine-tuned model, KGW performs better than TBW at $\tau = 0.7$ and is similar in trend to TBW at $\tau = 0.3$, suggesting its soft constraints are better tolerated by a model already adapted to the domain. In contrast, SynthID yields noticeably higher perplexity and wider distributions in the base and few-shot models, indicating

Model	Scheme	Samples Retained
Base	KGW	840
	SynthID	538
Few-shot	KGW	1000
	SynthID	1000
Fine-tuned	KGW	1000
	SynthID	1000

Table 6: Number of retained generations with perplexity ≤ 20 across model configurations, comparing KGW and SynthID.

reduced fluency and more frequent sampling of low-probability tokens. Only 538 base model generations were retained under the perplexity cap of 20. In the fine-tuned model, SynthID performs better, but still shows greater perplexity spread than KGW or TBW.

B.2.2 BERTScore Evaluation

We evaluate BERTScore F1 for generations produced with KGW and SynthID, using the same test setup and reference alignments as described in Section 4.1.2. Results are presented in Figure 6.

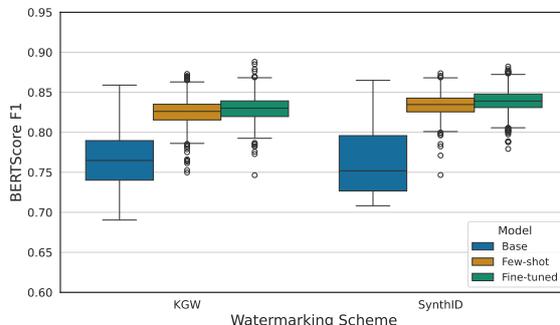


Figure 6: BERTScore F1 distributions across model configurations with KGW and SynthID. Higher values indicate greater semantic similarity to the human-written reference.

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 $\tau = 0.7$.

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.

C Robustness Evaluations

We provide additional details for the robustness evaluations described in Section 4.2. We include ROC curves for topic-based watermarking (TBW) and compare detection accuracy against the KGW and SynthID baselines under paraphrasing attacks. These results offer a more comprehensive view of how watermarking methods perform under realistic adversarial transformations.

C.1 ROC Curves

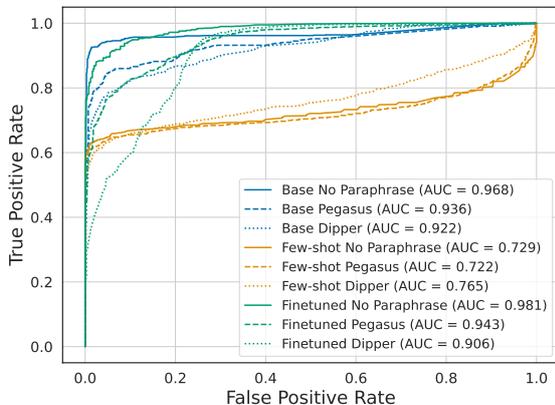


Figure 7: ROC curves for TBW detection under no attack, PEGASUS, and DIPPER paraphrasing, across all model configurations. The curves demonstrate TBW’s robustness across attack severity and adaptation settings.

Figure 7 presents ROC curves for TBW evaluated on outputs from the base, few-shot, and fine-tuned models. Detection performance remains strong in the base and fine-tuned settings, with area under the curve (AUC) values exceeding 0.90 under no attack and only moderately degraded under paraphrasing. The few-shot model is more sensitive to

topic dilution, as discussed in Section 4.2, resulting in lower recall and reduced detection confidence under attack conditions.

C.2 Baseline Watermarking Robustness

To assess detection robustness of baseline methods, we apply the same paraphrasing attacks (PEGASUS and DIPPER) to generations produced by KGW and SynthID, and then evaluate each method’s ability to recover the watermark. Each row in Table 2 reflects detection accuracy out of 1,000 watermarked samples per setting.

Language Model	Attacks	Detection Accuracy		
		TBW	KGW	SynthID
Base	No Attack	0.9460	0.9710	0.9090
	PEGASUS	0.8470	0.4770	0.1350
	DIPPER	0.8760	0.7540	0.1730
Few-shot	No Attack	0.6220	0.9750	0.9590
	PEGASUS	0.5800	0.5800	0.3590
	DIPPER	0.5170	0.7480	0.2250
Fine-tuned	No Attack	0.8800	0.9260	0.9600
	PEGASUS	0.5830	0.4370	0.1800
	DIPPER	0.5840	0.6570	0.1590

Table 7: Detection accuracy of TBW, KGW, and SynthID across model configurations and paraphrasing attack types. Each score reflects the proportion of correctly identified watermarked samples out of 1,000 examples per condition. Bolded values indicate the best result per row.

Under no-attack conditions KGW and SynthID outperform TBW in the few-shot and fine-tuned models. In the base model variant, TBW performs better than SynthID, but still worse than KGW with a smaller margin.

Under paraphrasing, TBW shows better robustness. In the base model, TBW outperforms KGW and SynthID by a wide margin, maintaining detection accuracy above 84% under PEGASUS and 87% under DIPPER. KGW degrades more sharply, and SynthID performs poorly across all paraphrasing conditions. In the few-shot setting, TBW and KGW perform similarly under PEGASUS, but TBW trails slightly under DIPPER. SynthID again suffers larger drops in accuracy. In the fine-tuned model, TBW maintains accuracy comparable to KGW and outperforming SynthID.

D Classifier Specifics

We provide implementation details for the classification experiments described in Section 4.3.1. We outline the training setup used for both BERT and RoBERTa classifiers and summarize the evaluation

strategy for attribution analysis on generated peer reviews.

D.1 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 include:

- **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 (for memory efficiency)
- **Evaluation strategy:** Per epoch; best model selected via F1 on validation set
- **Early stopping:** Enabled (patience = 1)

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.

D.2 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 8 and 9 present the confusion matrices for BERT and RoBERTa, respectively.

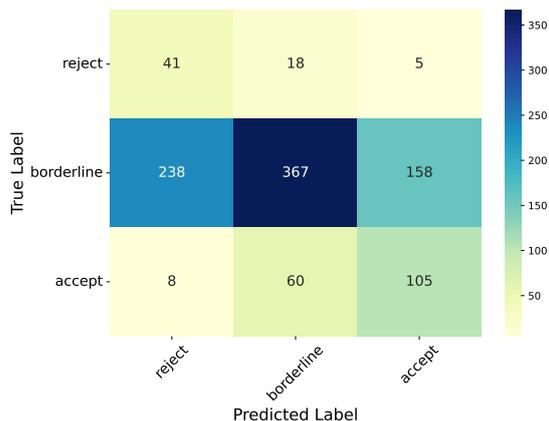


Figure 8: Confusion matrix for the BERT classifier on 1,000 human-written peer reviews.

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 accept labels, suggesting better overall discriminative performance.

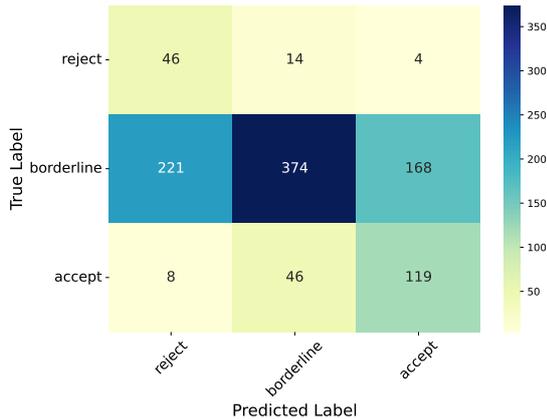


Figure 9: Confusion matrix for the RoBERTa classifier on 1,000 human-written peer reviews.

E Class-Specific Classifier Evaluation

To further characterize classifier performance, we conduct a class-specific evaluation of human-written peer reviews based on the same classification framework introduced in Section 4.3. This appendix extends the aggregate metrics reported in Table 3 by analyzing model behavior across the three target rating categories. Specifically, we examine confusion matrices for each classifier (BERT and RoBERTa), stratified by language model configuration (base, few-shot, fine-tuned) and watermarking condition (with or without topic-based watermarking). These matrices provide insight into the distribution of true versus predicted labels, allowing us to identify patterns of misclassification across rating levels.

Overall, we observe that classifier performance is strongest for the accept and borderline categories, with higher precision and recall scores relative to the reject class. This trend holds consistently across most configurations. The primary exception is observed in the BERT classifier applied to generations from the base LLM (without watermarking), where performance on the borderline class drops, leading to more frequent misclassifications into the neighboring categories.

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.

F Classifier-Based Attribution under Lower Topic Similarity Threshold ($\tau = 0.3$)

We extend our classifier-based attribution analysis to topic-based watermarking (TBW) applied at a lower semantic similarity threshold of $\tau = 0.3$, using the same evaluation methodology described in Section 4.3. This threshold relaxes the token-to-topic alignment constraints, thereby increasing green-list coverage and watermark signal strength, while potentially impacting semantic coherence.

Across classifiers and model variants, we observe a more balanced distribution of predictions among the three rating categories: accept, borderline, and reject. This suggests that the broader topic alignment may reduce overfitting to specific semantic patterns. However, in the fine-tuned model configuration, misclassifications of reject reviews remain more pronounced, indicating continued difficulty in capturing the linguistic signals associated with negative evaluations, even under stronger watermarking. The results are illustrated in Figure 10. Table 8 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.

G 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 12 presents accuracy changes across all classifier and model configurations. Table 9 reports the number of label transitions (e.g., Accept \rightarrow Borderline) observed in the paraphrased reviews. These metrics reflect the semantic resilience of reviewer intent and classification stability under

Table 8: 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

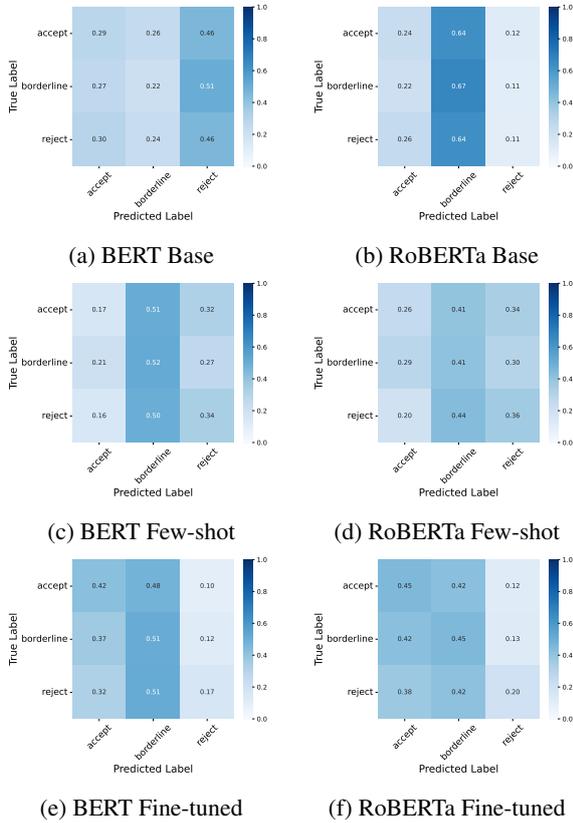


Figure 10: Confusion matrices for topic-based watermarking (TBW) applied at a lower topic similarity threshold ($\tau = 0.3$). Results are shown across all model configurations (base, few-shot, fine-tuned) and for both BERT and RoBERTa classifiers.

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.

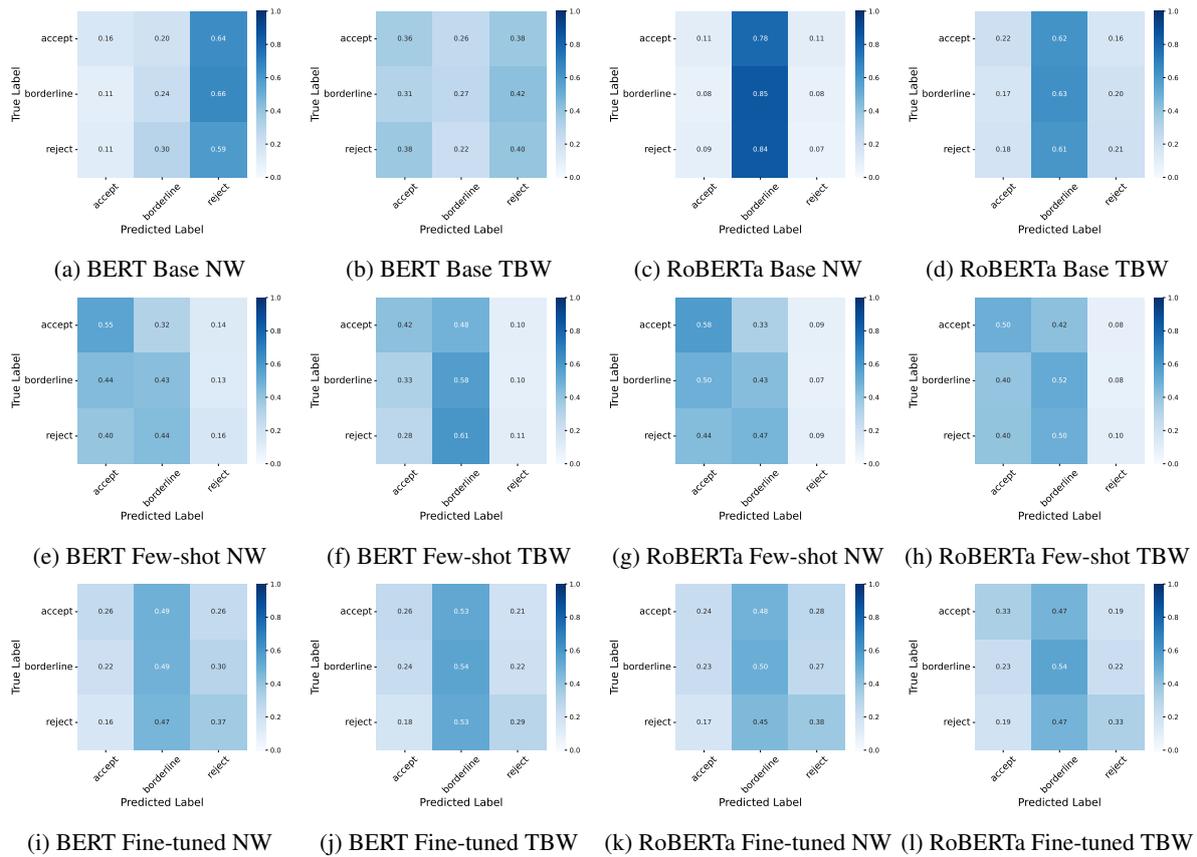


Figure 11: Confusion matrices comparing topic-based watermarking (TBW) at $\tau = 0.7$ with unwatermarked (NW) text across all model configurations. Each matrix reports the performance of either the BERT or RoBERTa classifier applied to outputs from three LLM variants: base, few-shot, and fine-tuned. Results highlight class-wise prediction behavior across watermarking and classifier settings.

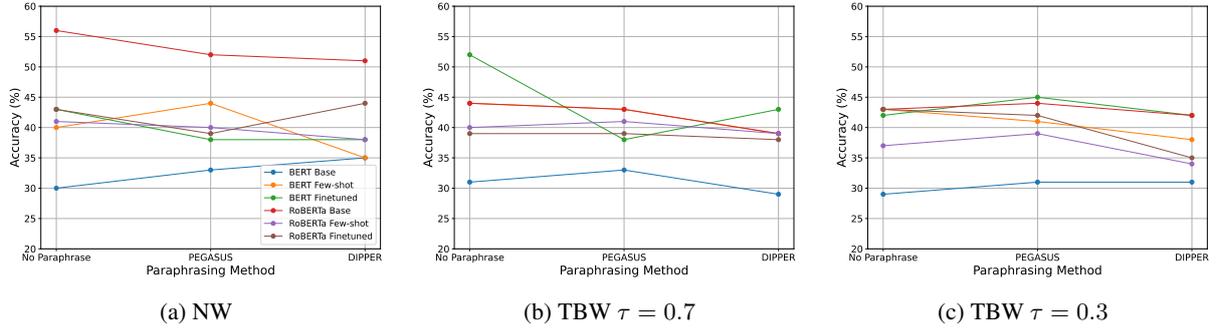


Figure 12: Classification accuracy on paraphrased peer reviews across three watermarking settings: (a) no watermark (NW), (b) topic-based watermarking (TBW) with $\tau = 0.7$, and (c) TBW with $\tau = 0.3$. Results are shown across all model configurations (base, few-shot, fine-tuned) for both BERT and RoBERTa classifiers under PEGASUS and DIPPER paraphrasing attacks.

Classifier	Model	Watermark	PEGASUS Shifts	DIPPER Shifts
BERT	Base	NW	58	54
		TBW-0.7	37	23
		TBW-0.3	51	45
	Few-shot	NW	24	14
		TBW-0.7	24	24
		TBW-0.3	24	22
	Fine-tuned	NW	27	20
		TBW-0.7	15	15
		TBW-0.3	25	15
RoBERTa	Base	NW	13	9
		TBW-0.7	23	25
		TBW-0.3	16	19
	Few-shot	NW	30	13
		TBW-0.7	27	22
		TBW-0.3	25	20
	Fine-tuned	NW	24	14
		TBW-0.7	18	22
		TBW-0.3	21	18

Table 9: 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-0.7, TBW-0.3), and evaluated separately under PEGASUS and DIPPER paraphrasing models.