Unsupervised Detection of LLM-Generated Text in Korean Using Syntactic and Semantic Cues

Heejeong Jeon[1] **Minsu Park**[1] **YunSeok Choi**[1] **Eunil Park**[1,2,*]
[1]Sungkyunkwan University, Seoul, Republic of Korea [2]Jaume I University, Castellon, Spain
{iamheej, mspark501}@g.skku.edu, {ys.choi, eunilpark}@skku.edu

Abstract

As Large Language Models (LLMs) are increasingly used for content creation, detecting AI-generated text has become a critical challenge. Prior work has largely focused on English, leaving low-resource languages such as Korean underexplored. We propose an unsupervised detection framework that integrates two complementary signals: syntactic token cohesiveness (TOCSIN) and semantic regeneration similarity (SimLLM). To support evaluation, we construct a Korean pairwise dataset of 1,000 anchors with continuation- and regeneration-style generations and further assess performance across domains (news, research paper abstracts, essays) and model families (GPT-3.5 Turbo, GPT-4o, HyperCLOVA X, LLaMA-3-8B). Without any training, our ensemble achieves up to 0.963 F1 and 0.985 ROC-AUC, outperforming baselines. These results demonstrate that the combination of syntactic and semantic cues enables robust unsupervised detection in low-resource settings. Code available at https://github.com/dxlabskku/llm-detection-main.

1 Introduction

Large Language Models (LLMs) (Achiam et al., 2023; Chowdhery et al., 2023; Touvron et al., 2023; Liu et al., 2024) have been widely adopted across various domains, including education, journalism, content creation, and scientific writing (Zellers et al., 2019; Ji et al., 2023). As their usage expands, so does the need to distinguish between human-written and AI-generated text, especially in high-stakes applications involving integrity, trust, and safety (Mitchell et al., 2023; Wu et al., 2025).

Early approaches relied on supervised classifiers trained on labeled datasets (Ippolito et al., 2020; Uchendu et al., 2020), but these often fail to generalize across LLMs, domains, or prompts and require retraining. To address these limitations, unsupervised methods exploit linguistic or statistical irregularities in the generated text (Mitchell et al., 2023; Ma and Wang, 2024), offering more flexibility and cross-lingual adaptability. However, most studies remain centered on English (Zellers et al., 2019), leaving low-resource languages like Korean underexplored (Park et al., 2025), where labeled resources are scarce. Recent cross-lingual efforts (Su et al., 2023) show promise, but performance is inconsistent without language-specific adaptations.

In addition to the scarcity of labeled resources, Korean may pose additional challenges for unsupervised LLM-generated text detection. As an agglutinative language with rich morphology and flexible word order, Korean can exhibit higher surface variability (Park et al., 2025), which may weaken the token-level regularities leveraged by the syntactic detectors (Ma and Wang, 2024; Su et al., 2023). Moreover, regeneration-based signals may be less stable in Korean, as minor morphological or functional variations can affect semantic consistency across regenerated outputs (Nguyen-Son et al., 2024; Krishna et al., 2023). These characteristics suggest that existing English-centric detection methods may not directly transfer to Korean without careful adaptation.

In this study, we propose **an unsupervised detection framework specialized for Korean**, adapting recent unsupervised detection techniques with Korean-specific modifications. We construct a novel pairwise dataset by prompting an LLM to generate both **continuation-style** and **regeneration-style** variants from human-written news summaries, yielding 1,000 anchors. While prior studies have explored continuation (Ma and Wang, 2024) and regeneration (Nguyen-Son et al., 2024) separately in English news, our work unifies both modes in a single Korean dataset, extending these approaches to low-resource languages. To validate robustness, we evaluate across diverse LLM

*Corresponding author.

Findings of the Association for Computational Linguistics: EACL 2026, pages 1504–1518
March 24-29, 2026

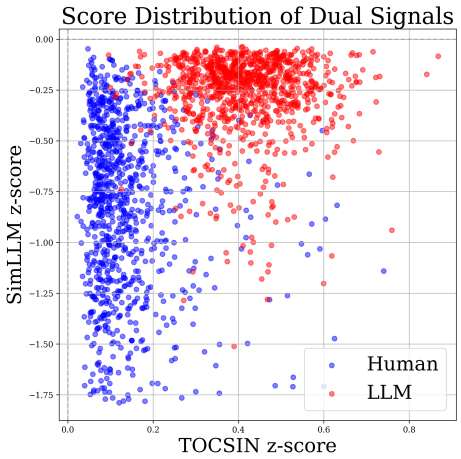

Figure 1: Score distribution of dual signals in the proposed framework. Each point represents a score using syntactic (TOCSIN) and semantic (SimLLM) cues.

families (GPT-3.5 Turbo, GPT-4o, HyperCLOVA X, LLaMA-3-8B) and domains (news, research paper abstracts, essays). We also provide supplementary experiments in Vietnamese, Thai, and Indonesian to explore portability beyond Korean.

To detect generated text, we combine two complementary signals: **(1) syntactic token cohesiveness (TOCSIN)**, which identifies anomalies by perturbing tokens and comparing the semantics of the original and masked versions (Ma and Wang, 2024); and **(2) semantic regeneration similarity (SimLLM)**, which measures semantic consistency by regenerating sentences and measuring the semantic similarity between the original and regenerated outputs (Nguyen-Son et al., 2024). These orthogonal cues are fused using ensemble strategies, including weighted-sum and sigmoid fusion, without additional training.

Experiments on our Korean dataset, evaluated using threshold-based classification, show that the ensemble achieves up to 0.963 F1 and 0.985 ROC-AUC, surpassing individual detectors and rule-based baselines. Importantly, our approach does not require labeled data or fine-tuning.

The contributions of this study are as follows:

- We construct a **novel Korean pairwise dataset**, with 1,000 anchors per domain (news, paper abstracts, and essays), where each human-written sentence is paired with multiple continuation- and regeneration-style generations, and further build parallel datasets of the same scale for Vietnamese, Thai, and Indonesian using an identical pipeline.
- We present an unsupervised LLM-generated text detection framework specialized for Korean, which integrates syntactic (TOCSIN) and semantic (SimLLM) signals through **ensemble detection to effectively capture complementary cues without classifier training.**
- Through extensive experiments, we demonstrate **strong and consistent detection performance** across multiple metrics, heterogeneous LLM families, and text domains, and further provide supplementary evidence of cross-lingual portability via preliminary evaluations on other low-resource languages.

2 Related Works

LLM-generated Text Detection. Recent studies on LLM detection methods can be categorized into (1) supervised detection method, (2) unsupervised detection method based on linguistic or statistical cues, and (3) watermarking-based approaches (Ji et al., 2023; Uchendu et al., 2020; Kirchenbauer et al., 2023). Supervised methods (Ippolito et al., 2020; Zellers et al., 2019; Uchendu et al., 2020) train discriminators on labeled datasets and achieve high accuracy within the same domain but struggle to generalize. Unsupervised approaches leverage intrinsic features without labeled data (Mitchell et al., 2023; Ma and Wang, 2024), making them more adaptable to low-resource or cross-lingual settings. Our study belongs to this category. Watermarking methods embed detectable patterns during generation (Kirchenbauer et al., 2023; Hou et al., 2024; Ren et al., 2024; Krishna et al., 2023; Chang et al., 2024), but they require access to the generation model, which is rarely feasible in practice (Kuditipudi et al., 2024; Christ et al., 2024).

Unsupervised and Hybrid Detection Methods. Unsupervised approaches range from simple heuristics such as surface statistics (Zellers et al., 2019; Uchendu et al., 2020), text length (Krishna et al., 2023), or perturbation-based loss curvature (Mitchell et al., 2023), to more advanced pairwise methods like **TOCSIN** (Ma and Wang, 2024) and **SimLLM** (Nguyen-Son et al., 2024). To overcome the limitations of single-signal detectors, recent studies emphasize hybrid or ensemble strategies (Liang et al., 2023; Ji et al., 2023; Uchendu et al., 2020), noting that individual cues capture only partial aspects of AI-generated text and are vulnerable to prompt engineering or domain shifts.

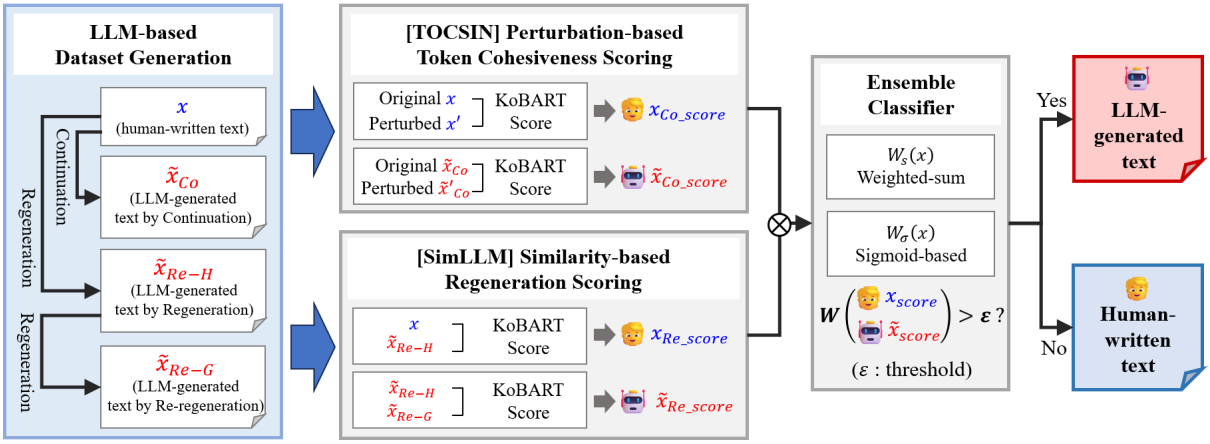

Figure 2: Overview of our unsupervised LLM-generated text detection framework. The notations **H** (human-originated) and **G** (generated-originated) indicate regeneration from human-written and LLM-generated texts, respectively. The leftmost input, labeled as *human-written text*, serves as the human reference in pairwise evaluation rather than the text under test. The final scores are computed for both human and LLM-generated inputs using $x_{\text{Co\_score}}, x_{\text{Re\_score}}$ and $\tilde{x}_{\text{Co\_score}}, \tilde{x}_{\text{Re\_score}}$.

Detection in Low-resource Languages. Although demand for multilingual detection is increasing, most research remains focused on English (Zellers et al., 2019; Su et al., 2023; Gehrmann et al., 2019), leaving low-resource languages such as Korean underexplored due to scarce datasets and limited benchmark corpora (Park et al., 2025). Recent studies investigate unsupervised methods with multilingual LLMs (Su et al., 2023); for example, Su et al. (2023) proposed log-rank-based detection but mainly evaluated English, while Park et al. (2025) highlighted linguistic challenges specific to Korean and the need for tailored feature engineering.

3 Dataset

We use the publicly available `daekeun-ml/naver-news-summarization-ko` dataset[1], which contains about 22,000 Korean news articles and summaries from July 2022. Following prior work such as TOCSIN and SimLLM (Nguyen-Son et al., 2024), we use the `summary` column to ensure comparable input length and style, filtering out sentences shorter than 20 tokens and removing stopwords and special characters.

We choose the Korean news dataset because of its high-quality human-written summaries, clear structure, and public availability, which supports reproducible and efficient experiments. To further test robustness across domains, we additionally include Korean *paper abstracts* and *essays*, following KatFishNet (Park et al., 2025), a similar study on Korean LLM detection that emphasized the importance of domain diversity.

Following the setting of TOCSIN (Ma and Wang, 2024), which employed 1,000 data points (500 human-written and 500 LLM-generated), our dataset consists of 4,000 samples: 1,000 human-written sentences and 3,000 LLM-generated counterparts. For each human-written sentence, we generate three variants using different sampling-based decoding strategies (e.g., temperature or top-p sampling) to ensure diversity and robustness. For each additional domain and cross-lingual dataset, the same pairing procedures are applied to ensure comparability across the experiments.

For clarity, each dataset is independently constructed with 1,000 anchors per domain or language. In particular, the Korean dataset consists of three domains (news, paper abstracts, and essays), each containing 1,000 anchors, while the Vietnamese, Thai, and Indonesian datasets each contain 1,000 anchors in the news domain.

Each human-written sentence acts as an anchor, paired with three derived LLM-generated sentences to form a single set. From this, we construct the following data types for pairwise comparisons: (1) human-written text, (2) LLM-generated text (for TOCSIN), (3) LLM-generated text (for SimLLM-H), and (4) LLM-generated text (for SimLLM-G). While the human-written text is taken directly from the original dataset and serves as the anchor across all pairs, we explain the remaining data types in the

[1] https://huggingface.co/datasets/daekeun-ml/naver-news-summarization-ko

Data Type	Human-written text (x)	LLM-generated text ($\tilde{x}_{Co}$)	LLM-generated text ($\tilde{x}_{Re-H}$)	LLM-generated text ($\tilde{x}_{Re-G}$)
Generation Method	–	Continuation ("Continue this text.")	Regeneration ("Revise this text.")	Re-regeneration ("Revise this text.")
Example	...on July 1, Korea's trade deficit in the first half of the year exceeded 10 billion dollars, marking a record-high.	...on July 1, analysts expect a continued decline in exports due to the global economic slowdown.	...on July 1, Korea recorded a trade deficit of over 10 billion dollars in the first half of the year, the highest level ever.	...on July 1, Korea recorded a trade deficit of over 10 billion dollars in the first half of the year, the highest level on record.

Table 1: Examples of human-written and LLM-generated texts used for each detection method.

following subsections. The overall structure of this pairing strategy is illustrated in Figure 2.

While our main experiments are conducted in Korean, we perform additional evaluations in other low-resource languages. We sample 1,000 Vietnamese sentences from a Vietnamese corpus (Hoa et al., 2025), 1,000 Thai sentences from the Thai Government Open Data corpus (ThaiGov v2)[2], and 1,000 Indonesian sentences from the `id_newspapers_2018` dataset[3]. All three datasets belong to the news domain, ensuring comparability with the Korean dataset, and the detailed cross-lingual experiments are provided in Appendix E.

3.1 LLM-generated Dataset for Syntactic Signal (Continuation)

For the TOCSIN method, we obtain continuation-style text by prompting the LLM to complete a truncated human-written sentence [Human → LLM (Continuation)]. Specifically, we use the first 20 tokens of each human-written sentence as a prefix and ask the model to continue it fluently. The Korean prompt is translated as: "*Complete the following Korean news sentence naturally and coherently. The final sentence should be approximately {target length} characters long. Below is the beginning of the sentence: {prefix}.*"

The prompt design follows the method introduced in TOCSIN (Ma and Wang, 2024). Unlike the original ending, the continuation may diverge in content, but TOCSIN focuses on token-to-token cohesiveness rather than semantic similarity, making it robust even when meanings differ. This typically induces stronger local cohesion in LLM outputs, which serves as a discriminative signal. An example of such generations is shown in Table 1.

[2] https://github.com/PyThaiNLP/thaigov-v2-corpus

[3] https://huggingface.co/datasets/indonesian-nlp/id_newspapers_2018

3.2 LLM-generated Dataset for Semantic Signal (Regeneration)

For the SimLLM method, we obtain regeneration-style text by prompting the LLM to carefully revise a human-written sentence while still preserving its meaning and natural sentence-final style [Human → LLM (Regeneration)]. The Korean prompt is translated as: "*Rewrite the following sentence to be more fluent and grammatically natural, while keeping the original meaning and sentence-final style unchanged. The sentence: {text}.*"

The prompt design follows the method introduced in SimLLM (Nguyen-Son et al., 2024). This process is applied twice: first to the human-written sentence [Human → LLM (Regeneration)], and then again to the regenerated output [LLM → LLM (Re-regeneration)]. The similarity gap between these pairs provides a semantic detection signal. This dual-generation setup captures both paraphrastic diversity and regeneration consistency. Examples of such generations are shown in Table 1.

4 Method

4.1 Detection Methods

We employ two unsupervised detection strategies: **syntactic token cohesiveness** (TOCSIN) and **semantic regeneration similarity** (SimLLM). Both methods originally relied on BARTScore (Yuan et al., 2021) as their similarity metric. To adapt them for Korean, we replace BARTScore with KoBARTScore[4], which is based on a KoBART model fine-tuned for Korean summarization. This adaptation ensures compatibility with Korean, preserving semantic fidelity and capturing stylistic cues.

[4] https://huggingface.co/digit82/kobart-summarization

4.1.1 TOCSIN: Perturbation-based Token Cohesiveness Scoring

As shown in the top branch of Figure 2, TOCSIN computes cohesiveness scores by comparing each sentence with its perturbed version. For a human-written text x, we randomly delete one token at a time to obtain a perturbed sentence x', and measure the drop in KoBARTScore between x and x', producing a human-side score x_{Co_score}. The same procedure is applied to an LLM-generated continuation $\tilde{x}_{Co}$, yielding its perturbed version $\tilde{x}'_{Co}$ and the score $\tilde{x}_{Co\_score}$. Higher average drops indicate stronger token dependencies, which are more characteristic of LLM outputs than human texts.

4.1.2 SimLLM: Similarity-based Regeneration Scoring

The bottom branch of Figure 2 illustrates SimLLM, which evaluates semantic stability through self-regeneration. For a human-written sentence x, we prompt the model to regenerate it into x_{Re-H} and compute the KoBARTScore similarity between them, resulting in a human-side score x_{Re_score}. For an LLM-generated sentence $\tilde{x}_{Re-H}$, we regenerate it to obtain $\tilde{x}_{Re-G}$ and then measure the similarity of this pair, producing the LLM-side score $\tilde{x}_{Re\_score}$. LLM-generated texts tend to be regenerated with minimal change, leading to higher similarity scores, whereas human-written texts typically undergo more diverse paraphrasing.

4.2 Score Ensemble

To integrate the syntactic and semantic detection signals, we design two ensemble z-score methods—weighted-sum and sigmoid-based scoring—that combine the normalized scores from TOCSIN and SimLLM to enhance robustness. For evaluation, the dataset is split into a *threshold tuning set* (800 samples) and an *evaluation set* (200 samples), where the former determines the optimal thresholds and the latter reports the performance. Unlike prior work, such as SimLLM, which allocated 10% of the data for testing, we adopt a 20% hold-out set to ensure robustness and reliability.

Weighted-sum Ensemble. We normalize the TOCSIN and SimLLM scores using z-score transformation to align their scales and then compute a weighted sum:

$$W_z(x_{\text{score}}, \tilde{x}_{\text{score}}) = \alpha \cdot x_{\text{score}} + (1-\alpha) \cdot \tilde{x}_{\text{score}} \quad (1)$$

The weight α controls the relative influence of each method; we vary it between 0.1 and 0.9 (step 0.05) to analyze the sensitivity and contribution of each signal, and select the value that yields the best detection performance (see Appendix B). The final weighted score is then thresholded for binary classification between human-written and LLM-generated texts.

Sigmoid-based Ensemble. As another ensemble approach, we average the normalized TOCSIN and SimLLM scores and apply a sigmoid function:

$$W_\sigma(x_{\text{score}}, \tilde{x}_{\text{score}}) = \sigma\big(\tfrac{1}{2}\big(x_{\text{score}} + \tilde{x}_{\text{score}}\big)\big) \quad (2)$$

This produces a probability score $W_\sigma(\cdot)$ between 0 and 1, then thresholded for binary classification into human-written or LLM-generated text.

4.3 Threshold Selection Strategies

After obtaining ensemble scores from TOCSIN and SimLLM, the final decision is made by applying a threshold ϵ to $W(x_{\text{score}}, \tilde{x}_{\text{score}})$, classifying the input as either human-written or LLM-generated. We emphasize that this thresholding step does not involve model training, but instead serves as a post-hoc, distribution-based calibration applied to fixed unsupervised scores. To determine ϵ, we consider two strategies using the tuning set.

(1) Threshold Sweep. We perform a linear sweep over the candidate thresholds and select the value that maximizes the F1-score on the tuning set. This provides a simple empirical method for identifying the best cutoff based on class score distributions.

(2) KDE-based Threshold Estimation. We estimate class-wise probability density functions (PDFs) for human and LLM scores using Gaussian kernel density estimation (KDE). In contrast to analytical methods that assume normality and rely on the mean and standard deviation, our implementation uses KDE as a non-parametric technique that directly fits the score distributions and determines the threshold at their first intersection, where both classes are equally probable. This non-parametric KDE yields stable unsupervised decision boundaries without requiring labeled validation data.

While our framework primarily operates on pairwise human–machine datasets for controlled evaluation, we also implement a pseudo-pairing extension that enables single-sentence inference by generating comparison texts in real time; details are provided in Appendix F.

5 Experiments

5.1 Experimental Setup

Datasets. For each domain or language, we independently constructed 1,000 pairwise sets of human-written and LLM-generated sentences following the procedure described in Section 3. To avoid overfitting to a single LLM, we generated texts from multiple model families: `GPT-3.5 Turbo` (Achiam et al., 2023) and `GPT-4o` (Achiam et al., 2023) from OpenAI, `HyperCLOVA X (HCX-005)` (Yoo et al., 2024) from Naver, and the open-source `LLaMA-3-8B` (Grattafiori et al., 2024) from Meta.

Scoring Models. For scoring in Korean, within our detection framework, we employed the pretrained KoBART checkpoint (`digit82/kobart-summarization`), chosen for its strong performance on Korean summarization tasks. Nevertheless, the framework is model-agnostic and KoBART can be replaced with any suitable Korean language model depending on the downstream objective or domain specificity. Additional experiments with alternative backbones such as KoBERT and Distil-KoBERT are reported in Appendix A.

In addition to Korean, we conducted cross-lingual evaluations to test the adaptability of our framework. Following the same pairing procedure and experimental protocol as in Korean, we replaced the scoring backbones with `BARTpho` (Tran et al., 2022) for Vietnamese, `WangchanBERTa` (Lowphansirikul et al., 2021) for Thai, and multilingual `mBART` (Liu et al., 2020) for Indonesian, while keeping all other hyperparameters identical. Further implementation details are provided in Appendix E.

Ensemble Scoring and Classification. Using the single-model detectors (TOCSIN and SimLLM), we first compute sentence-level scores and integrate them via ensemble strategies (Section 4.2), which then serve as the basis for binary classification. Binary classification thresholds were selected using two unsupervised strategies: sweep search and KDE-based estimation (Section 4.3). All hyperparameters, including α and thresholds, were tuned on the 800-set subset, and the random seed `42` was fixed for reproducibility. Following previous studies, we assessed the detection performance using classification metrics, including the F1-score, accuracy, and ROC-AUC.

For completeness, we present inference latency and token-based generation costs for both dataset construction and practical inference in Appendix G.

5.2 Baseline

For comparison, we implemented seven baselines in total: five rule-based methods from prior studies and two unsupervised detectors, TOCSIN and SimLLM. In addition, the baseline methods and ensemble models were evaluated on the unified dataset, which includes both continuation and regeneration generations. In contrast, the single-model evaluations of TOCSIN (continuation) and SimLLM (regeneration) were conducted on their respective task-specific subsets.

(1) **Entropy:** Predicts LLM if the LLM-generated sentence is longer than a human-written one, based on the intuition that LLMs tend to produce verbose outputs (Ippolito et al., 2020; Liang et al., 2023).

(2) **Logrank:** Computes the sum of Unicode character ranks in each sentence and labels as LLM if the sum is higher in the generated sentence (Mitchell et al., 2023; Park et al., 2025).

(3) **Likelihood:** Labels as LLM if the generated sentence contains more unique tokens than the human sentence, reflecting lexical variety (Mitchell et al., 2023; Zellers et al., 2019).

(4) **DetectGPT:** Detects LLM output by measuring the drop in log-probability between an input passage and its lightly perturbed variants, based on the observation that model-generated text lies in regions of negative log-probability curvature (Mitchell et al., 2023).

(5) **LRR:** Uses the Log-Likelihood Log-Rank Ratio (LRR), defined as the ratio between the summed token log-likelihood and log-rank under a reference language model, as a zero-shot detection score for LLM-generated text (Su et al., 2023).

(6) **TOCSIN and SimLLM (Sweep):** The original authors of each baseline method do not specify how thresholds are determined or employ a fixed threshold without tuning. To ensure a fair comparison, we apply sweep-based threshold selection to our tuning set and use the resulting thresholds for these baselines.

Method	GPT-3.5 Turbo			GPT-4o			HyperCLOVA X			LLaMA-3-8B		
	F1	ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1	ACC	AUC
Baseline												
Likelihood (Mitchell et al., 2023)	0.689	0.762	-	0.756	0.804	-	0.886	0.897	-	0.630	0.730	-
LogRank (Mitchell et al., 2023)	0.780	0.820	-	0.771	0.814	-	0.827	0.853	-	0.824	0.850	-
DetectLLM (LRR) (Su et al., 2023)	0.784	0.823	-	0.769	0.812	-	0.916	0.922	-	0.750	0.800	-
DetectGPT (Mitchell et al., 2023)	0.762	0.807	-	0.777	0.818	-	0.606	0.718	-	0.854	0.873	-
Entropy (Uchendu et al., 2020)	0.795	0.830	-	0.782	0.821	-	0.841	0.863	-	0.792	0.828	-
TOCSIN (Sweep) (Ma and Wang, 2024)	0.901	0.892	0.963	0.893	0.887	0.929	0.932	0.932	0.970	0.953	0.953	0.984
SimLLM (Sweep) (Nguyen-Son et al., 2024)	0.835	0.833	0.901	0.846	0.825	0.938	0.717	0.650	0.744	0.757	0.733	0.818
Ours - Single Model												
TOCSIN (KDE)	0.928	0.925	0.963	0.871	0.873	0.929	0.933	0.932	0.970	0.945	0.945	0.984
SimLLM (KDE)	0.838	0.833	0.901	0.867	0.863	0.938	0.645	0.675	0.744	0.745	0.738	0.818
Ours - Ensemble Model												
Weighted (Sweep)	**0.963**	**0.963**	0.983	0.915	0.915	0.961	0.937	0.938	**0.978**	0.955	0.955	**0.988**
Weighted (KDE)	**0.963**	**0.963**	0.983	0.917	0.917	0.961	**0.943**	**0.943**	**0.978**	**0.964**	**0.965**	**0.988**
Sigmoid (Sweep)	0.948	0.948	**0.985**	**0.945**	**0.945**	**0.976**	0.876	0.873	0.943	0.859	0.865	0.936
Sigmoid (KDE)	0.948	0.948	**0.985**	0.934	0.935	**0.976**	0.872	0.873	0.943	0.862	0.868	0.936

Table 2: Detection performance on GPT-3.5 Turbo, GPT-4o, HyperCLOVA X (HCX-005), and LLaMA-3-8B evaluation sets. The baselines produce binary outputs and do not support AUC. The results are based on separate evaluation sets with samples not used in training.

Method	Paper Abstract			Essay		
	F1	ACC	AUC	F1	ACC	AUC
Baseline						
TOCSIN (Sweep)	0.711	0.675	0.682	0.661	0.493	0.518
SimLLM (Sweep)	0.810	0.800	0.890	0.877	**0.877**	0.924
Ours - Single Model						
TOCSIN (KDE)	0.634	0.625	0.682	0.654	0.493	0.518
SimLLM (KDE)	0.837	0.825	0.890	**0.880**	**0.877**	0.924
Ours - Ensemble Model						
Weighted (Sweep)	0.811	0.825	**0.925**	0.861	0.849	**0.932**
Weighted (KDE)	**0.872**	**0.875**	**0.925**	0.831	0.822	**0.932**
Sigmoid (Sweep)	0.842	0.850	0.905	0.838	0.836	0.897
Sigmoid (KDE)	0.865	**0.875**	0.905	0.838	0.836	0.897

Table 3: Detection performance on Paper Abstracts and Essays generated with GPT-3.5 Turbo.

6 Results

The detection performances of our methods and the baselines are presented in Table 2.

Baseline Comparison. The rule-based baselines exhibit only moderate performance (F1 = 0.689–0.795 on GPT-3.5 Turbo), with entropy performing the best. They typically label a sentence as LLM-generated only under strong heuristic cues, often missing subtler instances. TOCSIN and SimLLM, evaluated here with sweep-based thresholds, outperform the rule-based baselines, especially TOCSIN, while SimLLM shows weaker performance in some cases. On GPT-3.5 Turbo and GPT-4o, they achieve F1 scores of 0.901/0.835 and 0.893/0.846, but their relative strengths vary across LLM families. For example, on HyperCLOVA X, TOCSIN remains strong (0.932) while SimLLM drops markedly (0.717). This variability indicates that single-model detectors can be effective in some cases but unreliable in others, making them less robust as standalone methods.

Single-Model Performance. As in the sweep-based baselines discussed above, TOCSIN generally achieves higher performance than SimLLM, though the advantage is not uniform across LLMs. On GPT-3.5 Turbo, TOCSIN reaches an F1 of 0.928 while SimLLM lags at 0.838, whereas on GPT-4o the gap narrows, with both methods achieving around 0.87. However, a different pattern emerges for non-OpenAI models: on HyperCLOVA X, TOCSIN maintains strong performance (F1 = 0.933), whereas SimLLM drops substantially (F1 = 0.645–0.717). These results indicate that while TOCSIN is a stable signal across diverse models, SimLLM can degrade sharply, especially on models outside the GPT family. Thus, single-model approaches risk being overly dependent on which signal happens to be stronger for a given LLM.

Ensemble-Model Performance. The ensemble methods address the weaknesses observed in single-model detectors by combining syntactic and semantic signals into a unified decision. Across all LLMs, the ensembles achieve the strongest or near-strongest performance—for example, the weighted ensemble reaches F1 = 0.963 on GPT-3.5 Turbo, and the sigmoid ensemble attains F1 = 0.945 with AUC = 0.976 on GPT-4o—consistently surpassing either TOCSIN or SimLLM alone. Even on HyperCLOVA X, where SimLLM drops to 0.717,

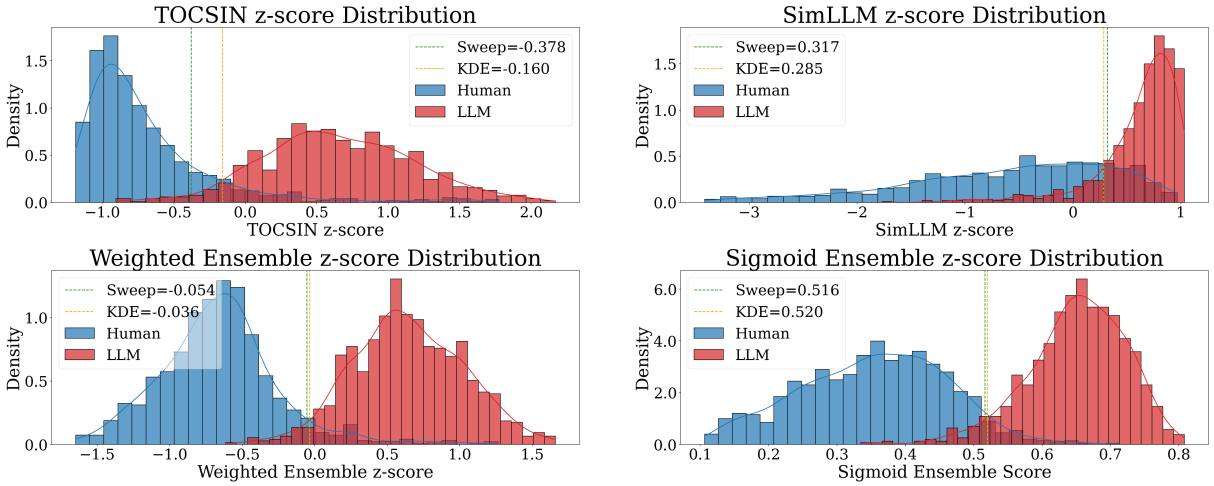

Figure 3: Histograms of detection scores for human vs. LLM-generated texts. The plots show the score distributions with decision thresholds. The y-axis shows probability density instead of raw frequency, which normalizes the histogram so the bars sum to 1.

the ensemble recovers to 0.943, showing that SimLLM's sensitivity complements TOCSIN's stability in combination. **Importantly, the ensemble remains robust regardless of which individual signal is weaker: whether SimLLM deteriorates on non-GPT models or TOCSIN loses strength in certain cases, the fusion strategy restores balanced detection capacity.** Overall, the ensembles demonstrate that, even without knowing in advance which signal may weaken, their integration ensures consistently strong and robust detection across diverse LLM families.

Domain and Cross-Lingual Generalization. Beyond model families, the ensemble strategies also generalize well across domains. As shown in Table 3, domain shift leads to fluctuations in single-model performance—for instance, in this case, unlike in the news domain, SimLLM shows stronger performance than TOCSIN in essays (0.880 vs. 0.654). Yet the ensemble consistently restores robust accuracy (Essay F1 = 0.831, Paper Abstract F1 = 0.872), demonstrating that even when one signal deteriorates, complementary fusion guarantees stable detection across heterogeneous text genres.

Beyond Korean, we further validated the framework on Vietnamese, Thai and Indonesian news datasets (details in Appendix E). The results confirm that while absolute performance is slightly lower due to language-specific challenges, the ensemble consistently outperforms single-signal baselines, reinforcing the portability of our approach across low-resource languages.

Visualization Analysis. Figure 3 illustrates the distribution of detection scores for human-written and LLM-generated texts. Among the visualizations, the sigmoid-based ensemble exhibits the clearest and most balanced class separation. This is attributed to the bounded output range (0 to 1) of the sigmoid function, which regularizes the score values and mitigates the impact of outliers. Although its F1 score and accuracy are slightly lower than those of the weighted ensemble, its interpretability and high ROC-AUC (0.985) make it suitable for real-world applications that require robust threshold calibration. Further analysis of the threshold selection for Sweep and KDE is discussed in the Appendix C.

To better understand the discriminative capacity of the proposed dual-signal detection framework, we visualize the score distribution of TOCSIN and SimLLM in a two-dimensional scatter plot (Figure 1). Notably, LLM-generated texts (red) are tightly clustered along the upper region of the SimLLM axis while occupying a broad range of lower TOCSIN values, whereas human-written texts (blue) tend to exhibit high syntactic fidelity (higher TOCSIN scores) but display greater variance in semantic alignment. This distribution confirms the near orthogonality of the two signals, enabling a robust separation in the joint feature space.

Qualitative Analysis. We analyze sentence samples based on Δ, defined as the absolute difference between the TOCSIN and SimLLM scores. A high Δ indicates that the two models make opposing pre-

dictions, suggesting ambiguity or conflicting cues in the data. These cases are prone to misclassification when using only one model, highlighting the benefits of ensemble methods for robust detection.

We include top-Δ examples from both classes (human vs. LLM) to illustrate these. The samples are listed in Table 6 in the Appendix D.

7 Conclusion

We propose an unsupervised detection framework for identifying LLM-generated Korean texts by combining syntactic token cohesiveness (TOCSIN) and semantic regeneration similarity (SimLLM), offering a solution suitable for low-resource languages. By capturing both structural irregularities and semantic deviations, our method provides robust detection signals across diverse text types. Importantly, our experiments demonstrate that while individual signals may fluctuate across LLM families or domains, their ensemble consistently restores balanced performance, underscoring the central role of fusion in achieving robustness. Furthermore, we validate the framework's portability across domains and additional low-resource languages, confirming adaptability beyond Korean. These findings highlight not only the methodological contribution but also the practical potential of our approach for reliable AI-generated text detection in multilingual and real-world settings.

8 Limitations

Our study has several limitations. First, our evaluation focuses on relatively formal text domains, including news articles, paper abstracts, and essays. As a result, informal or conversational genres such as social media posts and dialogue-based interactions are not covered. Extending the proposed framework to such domains remains an important next step toward broader real-world applicability.

Second, the pseudo-pairing mechanism currently relies on API-based LLM calls and domain-specific prompting strategies. Although this design enables single-text inference without additional training, it introduces dependencies on specific model interfaces and prompts. Future work will explore domain-agnostic pseudo-pairing strategies and open-source LLM-based implementations to improve reproducibility and scalability.

Finally, although we provide preliminary evidence of cross-lingual portability through experiments on Vietnamese, Thai, and Indonesian, these evaluations are limited in scale and domain coverage. A more comprehensive multilingual benchmark would be required to fully assess robustness across languages and text styles.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (RS-2025-25424137). This work was also supported by the MSIT (Ministry of Science and ICT), Korea, under the ICAN (ICT Challenge and Advanced Network of HRD) support program (RS-2024-00436934), and the ITRC (Information Technology Research Center) grant (RS-2024-00436936), supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. https://arxiv.org/abs/2303.08774.

Yapei Chang, Kalpesh Krishna, Amir Houmansadr, John Frederick Wieting, and Mohit Iyyer. 2024. PostMark: A robust blackbox watermark for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8969–8987, Miami, Florida, USA. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, and 48 others. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Miranda Christ, Sam Gunn, and Or Zamir. 2024. Undetectable watermarks for language models. In *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pages 1125–1139. PMLR.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

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. https://arxiv.org/abs/2407.21783.

Tran Thai Hoa, Tran Quang Duy, Khanh Quoc Tran, and Kiet Van Nguyen. 2025. Vifactcheck: A new benchmark dataset and methods for multi-domain news fact-checking in vietnamese. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(1):308–316.

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.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic detection of generated text is easiest when humans are fooled. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1808–1822, Online. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 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. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 27469–27500.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. 2024. Robust distortion-free watermarks for language models. https://arxiv.org/abs/2307.15593.

Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. 2023. Gpt detectors are biased against non-native english writers. *Patterns*, 4(7):100779.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. https://arxiv.org/abs/2412.19437.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Lalita Lowphansirikul, Charin Polpanumas, Nawat Jantrakulchai, and Sarana Nutanong. 2021. Wangchanberta: Pretraining transformer-based thai language models. https://arxiv.org/abs/2101.09635.

Shixuan Ma and Quan Wang. 2024. Zero-shot detection of LLM-generated text using token cohesiveness. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17538–17553, Miami, Florida, USA. Association for Computational Linguistics.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. DetectGPT: Zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 24950–24962. PMLR.

Hoang-Quoc Nguyen-Son, Minh-Son Dao, and Koji Zettsu. 2024. SimLLM: Detecting sentences generated by large language models using similarity between the generation and its re-generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22340–22352, Miami, Florida, USA. Association for Computational Linguistics.

Shinwoo Park, Shubin Kim, Do-Kyung Kim, and Yo-Sub Han. 2025. KatFishNet: Detecting LLM-generated Korean text through linguistic feature analysis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21189–21222, Vienna, Austria. Association for Computational Linguistics.

Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. 2024. A robust semantics-based watermark for large language model against paraphrasing. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 613–625, Mexico City, Mexico. Association for Computational Linguistics.

Jinyan Su, Terry Zhuo, Di Wang, and Preslav Nakov. 2023. DetectLLM: Leveraging log rank information for zero-shot detection of machine-generated text. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12395–12412, Singapore. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. https://arxiv.org/abs/2302.13971.

Nguyen Luong Tran, Duong Le, and Dat Quoc Nguyen. 2022. Bartpho: Pre-trained sequence-to-sequence models for vietnamese. In *Proc. Interspeech 2022*, pages 1751–1755.

Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. 2020. Authorship attribution for neural text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8384–8395, Online. Association for Computational Linguistics.

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

Kang Min Yoo, Jaegeun Han, Sookyo In, Heewon Jeon, Jisu Jeong, Jaewook Kang, Hyunwook Kim, Kyung-Min Kim, Munhyong Kim, Sungju Kim, and 1 others. 2024. Hyperclova x technical report. https://arxiv.org/abs/2404.01954.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: evaluating generated text as text generation. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, pages 27263–27277.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 9054–9065.

A Scoring Model Ablation (Korean)

To examine whether our framework is tied to a specific scoring backbone, we further experimented with alternative Korean models, `KoBERT`[5] and `Distil-KoBERT`[6]. As shown in Table 4, KoBART consistently provides the strongest performance, particularly under ensemble configurations where both weighted and sigmoid fusion reach F1 scores around 0.96. At the same time, even with weaker backbones such as KoBERT or Distil-KoBERT, the complementary nature of TOCSIN and SimLLM enables the ensemble to maintain stable detection accuracy. This confirms that while KoBART is the most effective choice for Korean semantic similarity, the framework itself remains robust and flexible across different scoring backbones.

Method	KoBERT			Distil-KoBERT		
	F1	ACC	AUC	F1	ACC	AUC
Baseline						
TOCSIN (Sweep)	0.893	0.887	0.929	0.891	0.887	**0.951**
SimLLM (Sweep)	0.846	0.825	0.938	0.680	0.570	0.682
Ours - Single Model						
TOCSIN (KDE)	0.871	0.873	0.929	**0.902**	**0.897**	**0.951**
SimLLM (KDE)	0.867	0.863	0.938	0.634	0.625	0.682
Ours - Ensemble Model						
Weighted (Sweep)	0.915	0.915	0.961	0.885	0.877	0.944
Weighted (KDE)	0.917	0.917	0.961	0.890	0.885	0.944
Sigmoid (Sweep)	**0.945**	**0.945**	**0.976**	0.804	0.792	0.885
Sigmoid (KDE)	0.934	0.935	**0.976**	0.815	0.812	0.885

Table 4: Ablation study with KoBERT and Distil-KoBERT as scoring backbones on Korean datasets.

B Effect of α (Weight Ratio)

In the weighted ensemble, α controls the relative contribution of TOCSIN (syntactic cohesiveness) and SimLLM (semantic similarity). We systematically varied α from 0.10 to 0.90 in increments of 0.05 to examine the sensitivity of detection performance. As shown in Table 5, performance remains robust across a wide range of α values, consistently exceeding F1 = 0.85. The best balance was achieved at $\alpha = 0.70$, which yielded the highest F1-score (0.963) under both sweep- and KDE-based thresholding. This indicates that, while TOCSIN tends to contribute more strongly in Korean due to its syntactic fidelity, the ensemble remains stable and competitive even when the weight is shifted toward SimLLM. Thus, the framework does not rely on a single fixed setting but instead benefits from adaptive tuning on the threshold set, ensuring robustness across datasets and languages. For each new domain, scoring model, or language, α is dynamically searched to find the optimal balance between syntactic and semantic signals, rather than being fixed to a constant value.

α	Sweep			KDE		
	F1	ACC	AUC	F1	ACC	AUC
0.10	0.852	0.850	0.933	0.852	0.850	0.933
0.15	0.870	0.868	0.946	0.873	0.873	0.946
0.20	0.886	0.882	0.955	0.884	0.885	0.955
0.25	0.888	0.892	0.963	0.910	0.910	0.963
0.30	0.909	0.907	0.970	0.912	0.912	0.970
0.35	0.932	0.935	0.975	0.925	0.925	0.975
0.40	0.921	0.920	0.980	0.930	0.930	0.980
0.45	0.933	0.932	0.983	0.942	0.943	0.983
0.50	0.934	0.932	0.985	0.950	0.950	0.985
0.55	0.957	0.958	0.986	0.953	0.953	0.986
0.60	0.956	0.955	0.985	0.956	0.955	0.985
0.65	0.961	0.960	0.984	0.961	0.960	0.984
0.70	**0.963**	**0.963**	**0.983**	**0.963**	**0.963**	**0.983**
0.75	0.956	0.955	0.982	0.958	0.958	0.982
0.80	0.936	0.932	0.980	0.958	0.958	0.980
0.85	0.958	0.958	0.978	0.951	0.950	0.978
0.90	0.938	0.935	0.973	0.939	0.938	0.973

Table 5: Sensitivity analysis of the weighted ensemble with varying α under Sweep and KDE thresholding.

C Effect of Threshold Selection

The optimal thresholds determined via KDE and Sweep vary by method and detection setting. For TOCSIN and SimLLM, the KDE-based thresholds are –0.160 and 0.285, respectively, while the corresponding Sweep-based thresholds are –0.378 and 0.317. In the ensemble models, the weighted variants yield thresholds of –0.036 (KDE) and –0.054 (Sweep), whereas the sigmoid-based variants use closely aligned values of 0.520 (KDE) and 0.516 (Sweep). These threshold values are consistent with the overall shape and relative scaling of the score distribution produced by each method, as observed in the visualization. Similarly, threshold selection is performed in an adaptive manner for each dataset, scoring backbone, and language by systematically searching the score distributions, allowing the framework to adjust to domain- and model-specific variations.

[5] https://github.com/SKTBrain/KoBERT

[6] https://github.com/monologg/DistilKoBERT

Top-Δ (Human)	**Top-Δ** (LLM)	**Low-Δ** (Human)	**Low-Δ** (LLM)
Misclassified (Human→LLM)	Misclassified (LLM→Human)	Correctly classified	Correctly classified
올해 연방준비제도 Fed · 연준 의 긴축 드라이브에 경기침체 우려가 확대되면서 올해 2분기 각종 원자재 가격이 하락했지만 공급 부족 해결이 쉽지 않기 때문에 원자재 가격이 비교적 높은 수준을 유지할 것으로 전문가들은 내다봤다.	최근 금융시장 변동성이 커지자, 이복현 금융감독원장은 여신전문금융회사에게 강도 높은 리스크 관리를 주문했고, 이를 위해 모든 프로젝트파이낸싱(PF) 대출이 적정하게 이루어졌는지 실태조사를 예고하며 리볼빙 서비스 불완전판매를 막기 위한 대책 마련을 지시했습니다.	헌법재판소의 '변호사 광고에 관한 규정' 부분위헌 판결로 인해 한국공법학회 회장 선정원이 오는 15일 HJ비즈니스센터 광화문점 세미나룸C 및 온라인 ZOOM에서 '온라인 플랫폼과 변호사 광고 규제'를 주제로 포럼을 개최하고 해당 결정이 리걸테크에 미치는 영향 혹은 시사점에 대한 학술적 논의가 필요하다는 인식에 따라 기획된 것으로 주제발표와 종합토론으로 진행될 예정이다.	국내 631개 주요 기업의 정보보호 등 ICT 투자지표가 공개되었는데, 단순한 정보보호 현황에 대한 점검뿐만 아니라 전체 IT 투자를 통해 디지털 시대 각 산업군에서 얼마나 많은 투자규모를 가져가고 있는 지 파악할 수 있는 토대가 마련되어, 매년 기업들의 IT 투자 규모를 확인할 수 있는 확인이 가능해졌다.

Table 6: Representative descriptions for each Δ-based category.

D Δ-based Analysis

As illustrated in Table 6, high-Δ samples represent edge cases in which TOCSIN and SimLLM assign contrasting judgments. In the first column (Top-Δ Human), the sentence contains excessively long noun phrases and repeated forecasting expressions. TOCSIN may misclassify such patterns as machine-generated owing to their structured repetition, whereas SimLLM captures the semantic flow and considers the sentence to be human-written. This results in a misclassification by TOCSIN, incorrectly labeling a human-written sentence as LLM-generated.

In the second column (Top-Δ LLM), the LLM-generated sentence is semantically rich and well-organized, prompting SimLLM to assign it a high LLM-likeness. However, its grammatical construction and phrasing resemble formal human-written press content, leading TOCSIN to assign it a low machine-likeness score. As a result, TOCSIN misclassifies this LLM-written sentence as human-written. This contrast highlights the orthogonal nature of the two signals: one focusing on the surface form and the other on meaning.

On the other hand, low-Δ samples demonstrate strong agreement between the two models. In the third column (Low-Δ Human), the sentence is a formal academic event announcement with stable grammar and semantics, leading both models to classify it as human-written text. Similarly, the fourth column (Low-Δ LLM) features a generation with numerical data and well-structured information, which both models agree is machine written.

In summary, high-Δ samples are often characterized by structural repetition, patterned phrasing, and semantic redundancy—factors that affect the two models differently. TOCSIN is sensitive to syntactic anomalies, whereas SimLLM is influenced by semantic coherence. Low-Δ samples, on the other hand, show model agreement and typically reflect grammatically and semantically stable content, either from humans or well-formed LLM outputs.

E Cross-lingual Evaluation

While our main study focuses on Korean as a representative low-resource language, we also conducted preliminary experiments on three additional languages—Vietnamese, Thai and Indonesian—to test whether the proposed framework can be extended beyond Korean. These experiments are not intended as a comprehensive multilingual benchmark, but as a step to verify portability across languages with distinct scripts and structures.

For these experiments, we sampled 1,000 sentences per language. For Vietnamese, we used a publicly available Vietnamese news corpus (Hoa et al., 2025) and employed `BARTpho` (Tran et al., 2022) as the scoring model. For Thai, we used the Thai Government Open Data corpus (ThaiGov v2)[7] with `WangchanBERTa` (Lowphansirikul et al., 2021). For Indonesian, we used the `id_newspapers_2018` dataset[8] and used the multilingual `mBART` model (Liu et al., 2020) for scoring.

[7] https://github.com/PyThaiNLP/thaigov-v2-corpus

[8] https://huggingface.co/datasets/indonesian-nlp/id_newspapers_2018

Method	Vietnamese			Thai			Indonesian		
	F1	ACC	AUC	F1	ACC	AUC	F1	ACC	AUC
Baseline (Sweep)									
TOCSIN (Sweep)	0.665	0.535	0.704	0.743	0.688	0.783	0.741	0.657	0.630
SimLLM (Sweep)	0.762	0.743	0.805	0.735	0.698	0.774	0.648	0.660	0.718
Ours - Single Model									
TOCSIN (KDE)	0.667	0.500	0.704	0.697	0.703	0.783	0.735	0.657	0.630
SimLLM (KDE)	0.771	0.760	0.805	0.669	0.507	0.774	0.667	0.500	0.718
Ours - Ensemble Model									
Weighted (Sweep)	0.776	0.772	**0.842**	0.768	0.748	**0.842**	**0.747**	0.667	0.698
Weighted (KDE)	**0.780**	**0.777**	**0.842**	0.669	0.507	**0.842**	0.733	0.672	0.698
Sigmoid (Sweep)	0.772	0.748	0.837	**0.774**	**0.757**	0.840	0.735	**0.718**	**0.772**
Sigmoid (KDE)	0.767	0.765	0.837	0.769	0.755	0.840	0.722	0.715	**0.772**

Table 7: Cross-lingual evaluation on Vietnamese, Thai, and Indonesian datasets. Sweep-based results serve as baselines, while KDE and ensemble methods represent our proposed approaches.

In all three cases, we followed the same continuation/regeneration pairing procedure as in Korean, generating 1,000 LLM counterparts with GPT-3.5 Turbo. All other experimental settings (z-score normalization, threshold selection, and hyperparameters) remained identical to the Korean experiments.

The results in Table 7 show that our framework achieves consistent detection capacity across these additional languages. For Vietnamese, ensemble methods yield F1 scores around 0.78; in Thai they reach up to 0.77 with balanced accuracy; and in Indonesian they attain around 0.75 with stable ROC-AUC. Although the absolute performance is lower than in Korean, the trends remain the same: single-model signals fluctuate, but ensemble fusion consistently restores robust performance.

Overall, these findings suggest that our method is not tied to Korean-specific features and can be applied to other low-resource languages with appropriate scoring models. We leave a full-scale multilingual benchmark to future work.

F Pseudo-Pairing for Single-Text Setting

Our framework is originally designed for pairwise comparison between human-written and LLM-generated sentences, which allows precise evaluation under controlled settings. However, in real-world applications, inputs often arrive as single texts without paired counterparts. To address this gap, we extend our method with a pseudo-pairing mechanism.

Given a single input sentence, the system generates comparison candidates using the same continuation and regeneration procedures as in our main setup. These pseudo-pairs enable the computation of TOCSIN and SimLLM scores even when no explicit human–machine pair is available. The resulting scores are then fused using the same ensemble strategies, producing a real-time prediction for whether the input is human- or LLM-generated.

This adaptation bridges the gap between controlled pairwise evaluation and practical single-sentence inference. We have also implemented this extension and updated the code to GitHub at `https://github.com/dxlabskku/llm-detection-main`.

G Inference Latency and Generation Cost

To improve clarity regarding the cost of our framework, we report the inference latency and token-based generation cost associated with constructing comparison texts for TOCSIN and SimLLM.

Generation Length and Latency. For dataset construction, each human-written anchor is paired with one continuation (for TOCSIN) and two regeneration-based outputs (for SimLLM). On average, continuation generations contain 227 characters (approximately 52 tokens), while regeneration outputs contain 165 characters (approximately 37 tokens). The average generation time is 2.82 seconds per continuation request and 2.93 seconds per regeneration request.

Total Dataset Construction Cost. Table 8 summarizes the total generation cost for constructing the full Korean dataset of 1,000 anchors across different LLMs. For commercial APIs, the overall cost remains modest, while open-source models incur negligible API cost.

Model	Total Generation Cost
GPT-3.5 Turbo	1.27 USD
GPT-4o	5.03 USD
HyperCLOVA X	2,515 KRW ($\approx$ 1.88 USD)
LLaMA-3-8B (open-source)	-

Table 8: Total cost for dataset construction (1,000 anchors).

Cost in Practical Inference. We emphasize that dataset construction is a one-time offline process. In real-world deployment under the pseudo-pairing setting, inference for a single input requires generating only one continuation and two regenerations on-the-fly. As a result, the practical runtime overhead and cost during inference are significantly smaller than those incurred during full dataset construction.