SenDetEX: Sentence-Level AI-Generated Text Detection for Human-AI Hybrid Content via Style and Context Fusion

Lei Jiang[1], Desheng Wu[1*], Xiaolong Zheng[2]

[1]University of Chinese Academy of Sciences, Beijing, China
[2]Institute of Automation, Chinese Academy of Sciences, Beijing, China
jianglei232@mails.ucas.ac.cn, dwu@ucas.ac.cn, xiaolong.zheng@ia.ac.cn

Abstract

Text generated by Large Language Models (LLMs) now rivals human writing, raising concerns about its misuse. However, mainstream AI-generated text detection (AGTD) methods primarily target document-level long texts and struggle to generalize effectively to sentence-level short texts. And current sentence-level AGTD (S-AGTD) research faces two significant limitations: (1) lack of a comprehensive evaluation on complex human-AI hybrid content, where human-written text (HWT) and AI-generated text (AGT) alternate irregularly, and (2) failure to incorporate contextual information, which serves as a crucial supplementary feature for identifying the origin of the detected sentence. Therefore, in our work, we propose **AutoFill-Refine**, a high-quality synthesis strategy for human-AI hybrid texts, and then construct a dedicated S-AGTD benchmark dataset. Besides, we introduce **SenDetEX**, a novel framework for sentence-level AI-generated text detection via style and context fusion. Extensive experiments demonstrate that SenDetEX significantly outperforms all baseline models in detection accuracy, while exhibiting remarkable transferability and robustness. Source code is available at https://github.com/TristoneJiang/SenDetEX.

1 Introduction

In recent years, LLMs have advanced rapidly, driving significant progress in natural language processing tasks. Their generated texts now rival human writing in fluency and coherence (Laskar et al., 2024). However, the widespread adoption of LLMs also carries potential risks. These include academic misconduct (Koike et al., 2024), the spread of misinformation (Yin et al., 2024), and the misuse of generated content (Abdali et al., 2024). Therefore, developing accurate and efficient AGTD approaches is crucial for ensuring content reliability and mitigating potential threats.

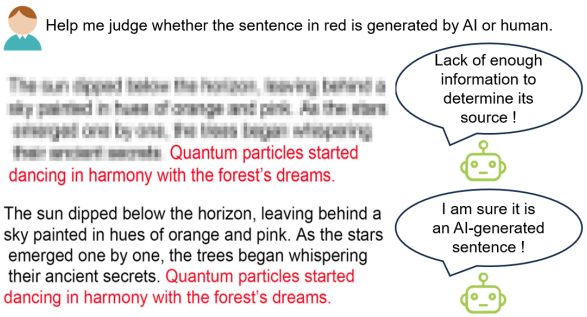

Figure 1: Contextual clues reveal text's authorship. The target sentence is grammatically correct and poetic, making it difficult to determine its origin solely based on its content. However, compared to the depiction of nature in the preceding sentences, its abrupt shift into abstract sci-fi imagery aligns more closely with the tendency of AI-generated text to blend "poetic + sci-fi" styles.

S-AGTD technology is proving increasingly valuable across multiple fields. Academic misconduct detection enables sentence-level analysis to identify AGT mixed with HWT. For news and content moderation, it helps assess the originality of AI-assisted texts. In legal and contract review, it detects AI-drafted clauses with potential risks or inconsistencies.

However, most existing AGTD works primarily focus on long texts represented by documents, while research on fine-grained analysis of short texts represented by sentences remains relatively scarce. Recent research highlights that the primary challenge in sentence-level text detection tasks stems from the insufficient stylistic cues resulting from the short text length, rendering reliable authorship attribution difficult (Zeng et al., 2024b). Directly transferring existing document-level AGTD (D-AGTD) methods to sentence-level texts is not feasible. For instance, training-free methods like DetectGPT (Mitchell et al., 2023) and DNA-GPT (Yang et al., 2024) rely on sufficient token length to

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 5287–5302
November 4-9, 2025

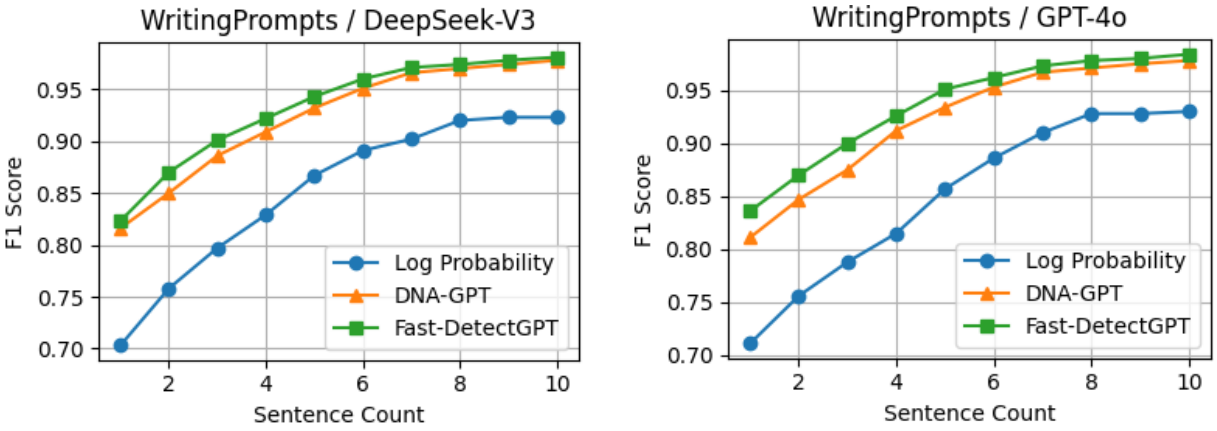

Figure 2: The relationship between sentence count and the detection performance of training-free detectors. The test data is sourced from WritingPrompts, and the AI-generated texts are created by DeepSeek-V3 and GPT-4o.

ensure the reliability of statistical metrics, while supervised methods such as Radar (Mao et al., 2024) and SCRN (Huang et al., 2024) struggle to capture stable and practical features in short texts and are prone to overfitting to the details of the training data.

Existing S-AGTD research also has certain limitations. For instance, the works in (Wang et al., 2023, 2024b) only discuss the boundary detection of author shifts under the "human-AI" writing paradigm, without considering scenarios like "AI-human," "human-AI-human," or other more complex human-AI text alternation patterns. Another work employs a pipeline of text segment separation and detection (Zeng et al., 2024b). Still, in mixed texts, frequent author shifts between adjacent sentences make it difficult for the segment detector to identify text segments with consistent authorship accurately. We carry out pre-experiments for three training-free AGTD methods (Solaiman et al., 2019; Bao et al., 2024; Yang et al., 2024) on the WritingPrompt subset (Fan et al., 2018)[1]. The results shown in Figure 2 reveal that detectors usually perform poorly when the input text length is very short. Besides, as sentence length increases, detection performance improves significantly.

Therefore, when the information the detected sentence provides is limited in identifying its authorship, our intuition is that the *preceding context can offer distinctive features to the short-text AGTD task*. This intuition finds theoretical support in **distributional semantics** (Lenci et al., 2008) from cognitive science, which posits that textual meaning is not determined by words in isolation but is constructed through contextual interaction. As shown in Figure 1, contextual clues can potentially reveal text authorship. Human language comprehension involves active meaning construction, utilizing background knowledge and context, rather than mechanical signal processing. Building on this cognitive framework, we conceptualize S-AGTD as a "computational construction process" that essentially models a text's "style" and "context", representing its intrinsic linguistic characteristics and the surrounding semantic environment.

For style modeling, following prior works (Xu et al., 2025; Wu et al., 2025), we employ token probability sequences and token entropy sequences to represent textual "precision" and "openness" respectively, noting that HWT typically exhibits lower average token probability and higher average entropy compared to AGT. For context, we consider not only the intrinsic semantics of the candidate sentence but also the inferred semantics from a regenerated sentence (sharing similar semantics with the candidate sentence, yet not identical) based on its preceding text. The inferred semantics reflect contextual cues, and prior works (Zhu et al., 2023; Mao et al., 2024) show that AGT typically exhibits a stronger coupling between intrinsic and inferred semantics than HWT.

In our work, we formally define the S-AGTD task as: given a multi-sentence document, determining whether each sentence is authored by a human or a specific LLM. We summarize our contributions as follows:

(1) We first propose AutoFill-Refine, a high-quality human-AI hybrid text synthesis strategy that combines the contextual awareness of fill-in models with the generative capability of autoregressive models, while ensuring more natural and

[1]Experimental details are provided in Appendix C.

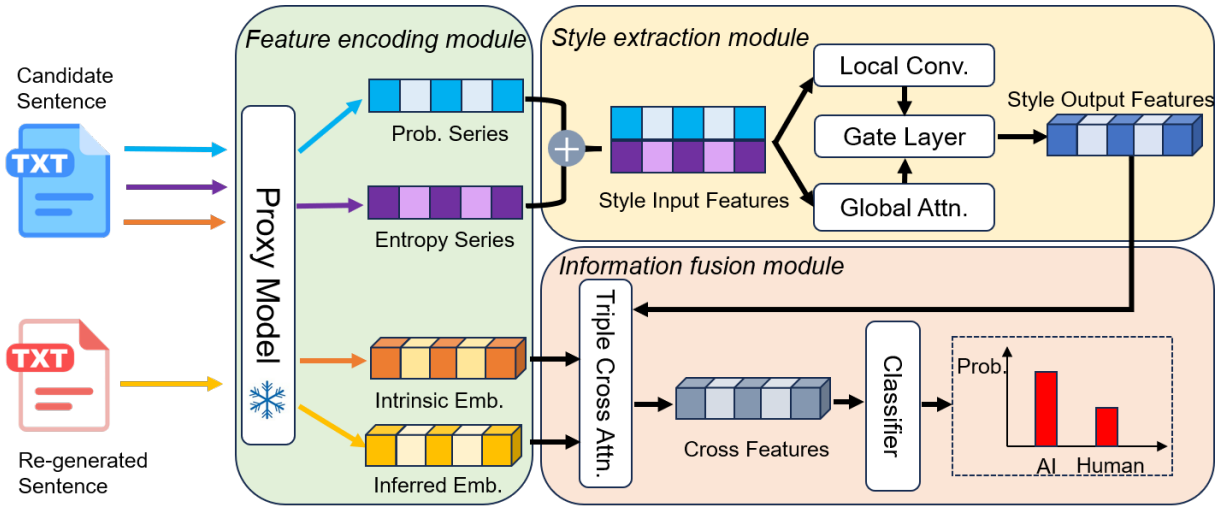

Figure 3: Workflow of the SenDetEX framework. "Prob.", "Emb.", "Conv.", "Attn." donote "Probability", "Embedding", "Convolution" and "Attention", respectively. The feature encoding module generates preliminary style and contextual representations. The style extraction module encodes the sequential probability and entropy signals into a dense style representation. The information fusion module integrates multi-source information to classify final sentence authorship attribution.

authentic text generation. This approach is used to construct our benchmark dataset, specifically designed for S-AGTD tasks.

(2) Recognizing that contextual information of target text provides potentially discriminative features for the short-text AGTD task, we propose SenDetEX - a novel S-AGTD framework that effectively models and integrates both "stylistic" and "contextual" information of target sentences for authorship attribution.

(3) Extensive experiments demonstrate that our proposed SenDetEX significantly outperforms other baselines in terms of detection performance across in-domain, cross-generator, and cross-domain scenarios, while exhibiting superior robustness against both word-level and sentence-level adversarial attacks.

2 Related Work

In the field of AGTD, one line of current research focuses on the tendency of LLMs to generate tokens with higher conditional probabilities at each position, indicating their higher "precision." Metrics such as perplexity (Hans et al., 2024) and log probability (Xu et al., 2025) have been employed as statistical indicators of AGT tendencies. The average likelihood of AGT is shown to decrease under perturbations (Mitchell et al., 2023), while token probability sequences are utilized as white-box features for supervised learning (Shi et al., 2024; Wang et al., 2023). Additionally, AGT often exhibits lower textual entropy, reflecting its lower "openness," which has been leveraged in text watermarking to identify watermark insertion locations (Wu et al., 2025; Liu and Bu, 2024) and dynamically adjust watermark weights (Lu et al., 2024). Furthermore, some studies argue that stylistic information is more effective than content information in characterizing authorship (Soto et al., 2024; Tripto et al., 2024). We employ time-series analysis methods to extract stylistic details embedded in the probability and entropy sequence signals.

Another line of research posits that AGT exhibits high similarity with its re-generated counterparts. Common-used methods for generating re-generated texts from the target text include revision (Zhu et al., 2023), paraphrasing (Mao et al., 2024), partial token deletion (Ma and Wang, 2024), multiple rounds of negation (Nguyen-Son et al., 2024), truncating portions of the text as prompts (Yang et al., 2024), and reconstructing prompts (Yu et al., 2024; Huang et al., 2025) for completion. In our approach, we use the preceding context of the target text as a prompt to obtain the re-generated texts. It will enable comparison of similarity with the target text, while also introducing richer contextual information. Moreover, contextual information has been shown to enhance the robustness of detectors against adversarial attacks (Hou et al., 2024a; Liu et al., 2024a; Hou et al., 2024b). Overall, our work

aims to construct a style-context-aware S-AGTD framework.

3 Proposed Method

Let s_i denote the i-th sentence of a document, our framework aims to predict whether s_i is generated by a human or a specific LLM. The re-generated sentence r_i of s_i is obtained by the white-box proxy model $\mathcal{M}_{\text{proxy}}$ with the c (serves as context-aware length) preceding sentences of s_i as the prompt[2]:

$$r_i = \mathcal{M}_{\text{proxy}}(\{s_{i-c-1}, \ldots, s_{i-1}\}),$$

The overall architecture is shown in Figure 3, which consists of three major modules: the **Feature Encoding Module**, the **Style Extraction Module**, and the **Information Fusion Module**.

3.1 Feature Encoding Module

Given a candidate sentence s_i and its re-generated counterpart r_i, we use a frozen proxy model $\mathcal{M}_{\text{proxy}}$ to extract token-level probabilistic signals and sentence-level embeddings.

Token Probability and Entropy Series. First, the proxy model computes the conditional probability of each token in s_i:

$$\mathbf{p}_i = \left(p(t_1|\cdot), p(t_2|\cdot), \ldots, p(t_{L_i}|\cdot)\right),$$

where L_i is the token length of s_i, and $p(t_j|\cdot)$ denotes the predicted probability of the ground-truth token t_j given the preceding context.

The entropy of each token prediction is defined as:

$$h_j = -\sum_{k=1}^{V} p_k^{(j)} \log p_k^{(j)},$$

where $p_k^{(j)}$ is the predicted probability for the k-th token in the vocabulary at position j, and V is the vocabulary size.

Thus, we obtain the entropy series:

$$\mathbf{e}_i = (h_1, h_2, \ldots, h_{L_i}) \in \mathbb{R}^{L_i \times 1}.$$

Sentence Embeddings. Additionally, we extract the intrinsic semantics embedding $\mathbf{z}_i^{\text{ins}}$ and the inferred semantics embedding $\mathbf{z}_i^{\text{inf}}$ by:

$$\mathbf{z}_i^{\text{ins}} = \mathcal{M}_{\text{proxy}}^{\text{embed}}(s_i) \in \mathbb{R}^{1\times d},$$
$$\mathbf{z}_i^{\text{inf}} = \mathcal{M}_{\text{proxy}}^{\text{embed}}(r_i) \in \mathbb{R}^{1\times d},$$

where $\mathcal{M}_{\text{proxy}}^{\text{embed}}(\cdot)$ denotes the sentence encoder module of the proxy model, and d is its embedding size.

[2]Following Appendix E when $c = 0$ or $i = 0$.

3.2 Style Extraction Module

Input Features. We first concatenate $\mathbf{p}_i$ and $\mathbf{e}_i$ along the feature dimension:

$$\mathbf{S}_i = \text{Concat}(\mathbf{p}_i, \mathbf{e}_i) \in \mathbb{R}^{L_i \times 2}.$$

Dual-Path Encoding. The concatenated feature $\mathbf{S}_i$ is processed by two parallel branches:

- **Local Branch.**

  We apply a dynamic depthwise separable convolution to extract local style information:

  $$\mathbf{S}_i^{\text{local}} = \text{DepthwiseConv1D}(\mathbf{S}_i, k = 5),$$
  $$\mathbf{S}_i^{\text{local}} = \text{PointwiseConv1D}(\mathbf{S}_i^{\text{local}}) \in \mathbb{R}^{L_i \times d},$$

  where the depthwise convolution (Howard, 2017) captures token-level local dependencies with kernel size $k = 5$, and pointwise convolution (Howard, 2017) then maps the features into $\mathbb{R}^{L_i \times d}$.

- **Global Branch.**

  We apply a sparsely connected Transformer (Child et al., 2019) encoder to extract global style information:

  $$\mathbf{S}_i^{\text{global}} = \text{SparseTransformer}(\mathbf{S}_i),$$

  where SparseTransformer applies two self-attention layers with eight heads, but restricts each token to only attend to its top-4 most relevant tokens based on attention scores. We adopt fixed sinusoidal positional encoding to preserve order information, and do not share parameters across attention heads. After Transformer encoding, a linear projection maps features into $\mathbb{R}^{L_i \times d}$.

Gated Fusion. The gating mechanism dynamically determines the contribution of local and global features at each position, enabling flexible adaptation across different sentence structures. Then we dynamically combine local and global information:

$$\mathbf{g}_i = \sigma(\mathbf{W}_g[\mathbf{S}_i^{\text{local}}; \mathbf{S}_i^{\text{global}}]) \in \mathbb{R}^{L_i \times d},$$
$$\mathbf{S}_i^{\text{style}} = \mathbf{g}_i \odot \mathbf{S}_i^{\text{local}} + (1 - \mathbf{g}_i) \odot \mathbf{S}_i^{\text{global}} \in \mathbb{R}^{L_i \times d},$$

where $\mathbf{W}_g \in \mathbb{R}^{2d\times d}$ is a learnable linear projection, $\sigma(\cdot)$ is the sigmoid activation, and $\odot$ denotes element-wise multiplication.

Finally, we perform average pooling over the sequence length:

$$\mathbf{z}_i^{\text{style}} = \text{AvgPool}(\mathbf{S}_i^{\text{style}}) \in \mathbb{R}^{1\times d}.$$

Source	Generator	Doc. count	[MASK] Rate (%)	Sen. Count (train/valid/test)	Subset Notification
XSUM	DeepSeek-V3	3162	31.6	34989/11570/11763	XD
	GPT-4o	3248	32.2	37255/12078/12297	XG
WritingPrompts	DeepSeek-V3	1043	32.1	29009/9456/9075	WD
	GPT-4o	1105	32.9	29794/9864/10089	WG

Table 1: Statistics of the human-AI hybrid text dataset synthesized by the AutoFill-Refine strategy in this work. "Doc." and "Sen." denote "document" and "sentence", respectively. "[MASK] Rate" corresponds to the actual proportion of LLM-generated sentences. For each subset, the documents are randomly partitioned into training/validation/test sets at ratios of 60%, 20%, and 20% respectively. We use the combination of "source" and "generator" as the "subset notification". For example, XD denotes "XSUM & DeepSeek-V3".

3.3 Information Fusion Module

Triple Cross Attention. We fuse the style and semantic embeddings using a triple cross-attention module (Misra et al., 2021; Wang et al., 2024a):

$$\mathbf{Q}_1 = \mathbf{z}_i^{\text{style}}, \quad \mathbf{K}_1 = \mathbf{z}_i^{\text{ins}}, \quad \mathbf{V}_1 = \mathbf{z}_i^{\text{inf}},$$
$$\mathbf{Q}_2 = \mathbf{z}_i^{\text{ins}}, \quad \mathbf{K}_2 = \mathbf{z}_i^{\text{inf}}, \quad \mathbf{V}_2 = \mathbf{z}_i^{\text{style}},$$
$$\mathbf{Q}_3 = \mathbf{z}_i^{\text{inf}}, \quad \mathbf{K}_3 = \mathbf{z}_i^{\text{style}}, \quad \mathbf{V}_3 = \mathbf{z}_i^{\text{ins}},$$

where each query attends to the other two features.

The attention outputs are computed as:

$$\mathbf{a}_1 = \text{MultiHeadAttn}(\mathbf{Q}_1, \mathbf{K}_1, \mathbf{V}_1),$$
$$\mathbf{a}_2 = \text{MultiHeadAttn}(\mathbf{Q}_2, \mathbf{K}_2, \mathbf{V}_2),$$
$$\mathbf{a}_3 = \text{MultiHeadAttn}(\mathbf{Q}_3, \mathbf{K}_3, \mathbf{V}_3),$$

where each MultiHeadAttn (Vaswani et al., 2017) uses 4 heads, with per-head dimension $d/4$.

The final cross features are:

$$\mathbf{z}_i^{\text{cross}} = \text{Concat}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)\mathbf{W}_{\text{fusion}} \in \mathbb{R}^{1\times d},$$

where $\mathbf{W}_{\text{fusion}} \in \mathbb{R}^{3d\times d}$ is a linear projection layer.

Classification. Then $\mathbf{z}_i^{\text{cross}}$ is fed into a classifier to predict the probability of being AI-generated:

$$p_i = \sigma(\mathcal{F}_{\text{cls}}(\mathbf{z}_i^{\text{cross}})),$$

where $\mathcal{F}_{\text{cls}}$ is a fully connected layer with weight matrix $\mathbf{W} \in \mathbb{R}^{d\times 1}$ and bias $b \in \mathbb{R}$, and $\sigma(\cdot)$ denotes the sigmoid activation. The final binary prediction label $\hat{y}_i = \mathbb{I}[p_i > 0.5]$.

Training Objective. The model is optimized using Mean Squared Error (MSE) loss for binary classification:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(p_i - y_i)^2,$$

where $y_i \in \{0, 1\}$ is the ground-truth label.

4 Experiments and Main Results

4.1 Dataset Construction

Mask-filling models (e.g., BERT (Kenton and Toutanova, 2019), RoBERTa (Liu et al., 2019)) demonstrate superior contextual awareness, while advanced autoregressive models (e.g., GPT series (Hurst et al., 2024), Gemini (Team et al., 2023)) exhibit remarkable generative capabilities. We therefore propose combining their strengths through prompt engineering, with a data filtering mechanism to ensure synthesis quality, ultimately forming the "AutoFill-Refine" human-AI hybrid text synthesis strategy[3]. The detailed workflow is as follows:

1. **Original Data Selection:** Select benchmark corpus from human-written datasets, denoted as $D_{\text{ori}} = \{d_1, d_2, ..., d_m\}$, where d_i represents a document containing multiple coherent sentences and m is the total number of documents.

2. **Random Masking:** Perform random sentence masking by replacing a proportion γ of sentences in D_{ori} with [MASK] placeholders, generating the masked dataset $D_{\text{mask}} = \{d'_1, d'_2, ..., d'_m\}$.

3. **Generation:** Use structured prompts to drive the autoregressive model $\mathcal{M}_{\text{gen}}$ for completion tasks. The prompt template is:

   "Fill in each `[MASK]` in the following document with a single sentence to ensure overall fluency, coherence, and logic. Original document: d'_i. New completed document:"

   The generated results are denoted as $D_{\text{re}} = \{d''_1, d''_2, ..., d''_m\}$.

[3] A synthesis sample is shown in Appendix F.

Method	XD→XD			XG→XG			WD→WD			WG→WG		
	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC
Log Entropy	65.9	71.1	31.0	67.4	70.1	30.4	64.0	69.2	29.8	65.7	71.0	31.4
Log Probability	72.3	76.7	41.2	71.8	76.2	40.5	70.8	75.5	40.0	70.4	75.3	39.4
DNA-GPT	81.4	85.2	55.4	81.8	85.7	56.3	80.9	84.8	54.4	80.2	83.9	54.1
Fast-DetectGPT	84.0	88.0	59.6	84.7	88.5	60.5	81.7	85.6	57.7	83.8	87.5	59.1
SeqXGPT	91.2	93.6	69.8	91.5	94.0	70.5	91.8	94.3	71.1	90.6	93.2	69.5
POGER	92.7	95.2	72.1	93.1	95.6	72.6	90.2	92.8	69.0	92.2	94.7	71.6
SenDetEX	**97.4**	**98.6**	**79.5**	**97.7**	**98.8**	**80.0**	**96.8**	**98.1**	**79.0**	**97.1**	**98.3**	**79.2**

Table 2: The overall detection results under the in-domain scenario.

Method	XD→XG			XG→XD			WD→WG			WG→WD		
	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC
SeqXGPT	88.4	91.2	66.9	89.0	91.7	67.5	89.1	91.8	68.1	88.1	90.9	66.7
POGER	89.6	92.4	68.8	90.2	92.9	69.4	87.6	90.4	66.2	89.5	92.2	68.6
SenDetEX	**96.0**	**97.5**	**77.7**	**96.2**	**97.7**	**78.0**	**95.3**	**96.9**	**77.2**	**95.5**	**97.2**	**77.4**

Table 3: The detection results under the cross-generator scenario.

4. **Quality Filtering:** Compute perplexity (PPL) using an oracle model, retaining only documents in D_{re} that satisfy $\text{PPL}(d_i'') < \text{PPL}(d_i)$.

Following the works in (Bao et al., 2024; Xu et al., 2025), we select XSUM (Narayan et al., 2018) and WritingPrompts (Fan et al., 2018) as benchmark corpora. We employ DeepSeek-V3 (Liu et al., 2024b) and GPT-4o (Hurst et al., 2024) as $\mathcal{M}_{\text{gen}}$ (temperature set to 0.7), and LLaMA-3-8B (Grattafiori et al., 2024) as the oracle model, with m set to 5,000 and 1,500 respectively and $\gamma = 0.35$. Except for the temperature, we adopt the default API parameters, and the maximum number of attempts is set to 10 when generating sentences that meet the "quality filtering" requirements. Besides, we use WordNet (Miller, 1995) for sentence segmentation of the text. The statistics of the final synthesized dataset are shown in Table 1.

4.2 Experiment Settings

We select four representative training-free methods—Log Probability (Solaiman et al., 2019), Log Entropy (Gehrmann et al., 2019), DNA-GPT (Yang et al., 2024), and Fast-DetectGPT (Bao et al., 2024)—along with two supervised baselines closely related to our work, SeqXGPT (Wang et al., 2023) and POGER (Shi et al., 2024), with detailed descriptions and configurations provided in Appendix A. To mitigate distribution shifts and enhance generalization, we follow the work in (Zeng et al., 2024a) to implement a fine-tuned LLaMA2-7B model (Touvron et al., 2023) as $\mathcal{M}_{\text{proxy}}$, with fine-tuning details presented in Appendix B. The temperature of $\mathcal{M}_{\text{proxy}}$ is set to 0.7 during its generation process.

In line with the work in (Cornelius et al., 2024), we adopt AUC, F1, and MCC (Matthews Correlation Coefficient) as evaluation metrics for the S-AGTD task. To assess the robustness of the detectors, we follow the work in (Pan et al., 2024) and implement four adversarial attack strategies: random deletion, random substitution, paraphrasing, and back-translation, with implementation details in Appendix D.

SenDetEX is trained by the AdamW optimizer (Loshchilov and Hutter, 2017) with a learning rate of 1×10^{-4} and a weight decay of 0.01. We train the model for up to 50 epochs, applying early stopping with a patience of 5 based on the validation loss. The batch size is set to 32, and the context-aware length c is default to 3 (further discussed in Section 4.3.1). All experiments are conducted on two NVIDIA A100 GPUs.

4.3 Main Results

4.3.1 In-domain Detection

In this paper, we use the notation "X→Y" to indicate that the training set of "X" is used for training (supervised methods), and the test set of "Y" is

Method	XD→WD			WD→XD			XG→WG			WG→XG		
	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC
SeqXGPT	87.1	89.8	65.4	88.9	91.6	67.8	86.4	89.2	64.8	86.8	89.6	65.4
POGER	88.5	91.2	67.3	87.6	90.3	65.9	87.7	90.4	66.6	88.2	90.9	67.2
SenDetEX	**95.0**	**96.5**	**76.7**	**95.2**	**96.7**	**77.1**	**94.5**	**96.1**	**76.2**	**94.8**	**96.3**	**76.5**

Table 4: The detection results under cross-domain scenario.

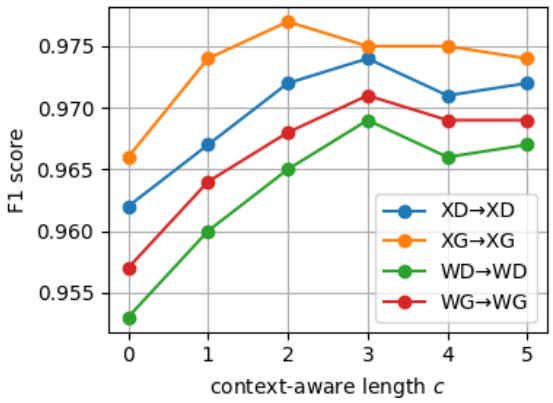

Figure 4: The relationship between context-aware length and in-domain detection performance.

used for evaluation. We first conduct in-domain AGTD evaluations on the XD, XG, WD, and WG datasets, where the source and target domains share the same source and generator. The results are shown in Table 2. The results demonstrate that our proposed SenDetEX consistently achieves the best in-domain AGTD detection performance across all datasets. Compared with the best-performing baselines, SenDetEX improves the F1 score by 4.7%–5.0%, AUC by 3.2%–3.8%, and MCC by 7.4%–7.8%, respectively.

The context-aware length c is a critical hyperparameter in SenDetEX, which controls the quality of the re-generated sentences. We set c to 0, 1, 2, 3, 4, and 5, and re-conduct the above in-domain experiments. The relationship between detection performance and c is illustrated in Figure 4. The results show that when c is small ($c < 3$), increasing c leads to better F1 scores, indicating that moderate context provides valuable semantic information. However, when c exceeds 3, further increases do not lead to significant performance improvement. The preceding text directly influences the generation of the regenerated sentence, which in turn affects the inferred embedding. The subsequent impact on the prediction performance of the SenDetEX framework is implicit and complex. Therefore, a larger c does not necessarily guarantee stable or theoretically supported improvements in detection accuracy. Based on the above analysis, we recommend setting $c = 3$, under which the XD→XD, WD→WD, and WG→WG experiments achieve the best detection performance.

4.3.2 Out-of Distribution Detection

In our work, we investigate two types of out-of-distribution (OOD) detection scenarios: cross-generator and cross-domain. These scenarios refer to cases where the generator or text domain of the target domain text differs from that of the source domain, respectively. The results of the cross-generator experiments are shown in Table 3. Our proposed SenDetEX consistently achieves the best detection performance across all four experiment groups. Specifically, for XD→XG, XG→XD, WD→WG, and WG→WD, SenDetEX outperforms the best baseline by 6.0%–6.4% in F1, 4.8%–5.1% in AUC, and 8.7%–9.1% in MCC.

The results of the cross-domain experiments are presented in Table 4. Although SenDetEX experiences a slight performance drop under cross-domain settings compared to the cross-generator scenarios, it still significantly outperforms all baselines. For XD→WD, WD→XD, XG→WG, and WG→XG, SenDetEX shows improvements of 6.6%–6.8% in F1, 5.1%–5.7% in AUC, and 9.3%–9.5% in MCC over the best baseline.

Based on the above discussion, SenDetEX demonstrates strong transferability in S-AGTD tasks and can quickly adapt to new environments. We attribute this capability to the coupling between the detected text's intrinsic semantics and its regenerated counterpart's inferred semantics, which provides generalizable and domain-invariant discriminative features for authorship distribution.

4.3.3 Robustness Study

The robustness of the AGTD method refers to the detector's ability to correctly identify the origin of

Method	random deletion			random substitution			paraphrasing			back-translation		
	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC
Log Entropy	61.5	67.4	27.4	59.8	65.6	26.0	54.9	60.9	22.9	53.4	59.4	21.8
Log Probability	69.0	73.7	37.9	67.5	72.3	36.7	63.6	68.6	34.4	62.5	67.4	33.5
DNA-GPT	79.1	83.0	52.4	78.9	82.6	53.8	74.0	78.6	48.3	73.2	77.6	47.3
Fast-DetectGPT	80.0	83.8	54.7	77.6	81.6	51.4	77.2	81.2	52.4	76.3	80.3	51.8
SeqXGPT	88.6	91.3	67.1	88.0	90.7	66.5	85.8	88.7	64.6	86.4	89.5	66.0
POGER	89.9	92.6	69.0	89.2	92.0	68.3	87.2	90.1	66.6	85.1	88.0	64.0
SenDetEX	**95.6**	**97.1**	**77.4**	**95.2**	**96.7**	**76.9**	**93.9**	**95.7**	**75.6**	**93.5**	**95.3**	**75.2**

Table 5: The overall detection results under adversarial attack scenario on XD→XD.

Method	XD→XD			XD→XG			XD→WD			paraphrasing		
	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC	F1	AUC	MCC
SenDetEX	97.4	98.6	79.5	96.0	97.5	77.7	95.0	96.5	76.7	93.9	95.7	75.6
-E	96.1	97.6	77.8	94.7	96.3	75.8	93.8	95.3	75.0	92.5	94.6	73.8
-R	95.6	97.3	77.2	92.6	94.9	73.1	91.6	93.6	72.0	90.0	92.5	70.5
-C	94.7	96.7	75.8	91.6	94.1	71.7	90.4	92.8	70.5	88.8	91.5	68.9
-E-R	93.5	95.7	73.4	90.4	93.0	70.0	89.3	91.8	68.5	88.1	90.8	67.8
-E-C	92.2	94.9	71.8	89.2	92.1	68.6	87.7	90.7	67.2	85.9	89.2	65.5

Table 6: Ablation Study on four S-AGTD scenarios. "-E": Removes entropy information by duplicating $\mathbf{p}_i$ as $\mathbf{e}_i$. "-R": Removes inferred semantics by duplicating $\mathbf{z}_i^{\text{ins}}$ as $\mathbf{z}_i^{\text{inf}}$. "-C": Removes both intrinsic and inferred semantics by replacing $\mathbf{z}_i^{\text{cross}}$ with direct input of $\mathbf{z}_i^{\text{style}}$ into $\mathcal{F}_{\text{cls}}$.

the input text even when it is subjected to adversarial perturbations. We redeploy our experiments on the XD→XD setting, and the results are presented in Table 5. Under word-level attacks, the performance of SenDetEX is barely affected. Compared to the non-adversarial setting shown in Table 2, the F1, AUC, and MCC scores under random deletion decrease by only 1.8%, 1.5%, and 2.1%, respectively. Under random substitution, the F1, AUC, and MCC scores drop by just 2.2%, 1.9%, and 2.6%, respectively, still outperforming other baselines. Compared to the best-performing baseline, SenDetEX achieves improvements of 6.7%, 5.6%, and 9.0% in F1, AUC, and MCC under paraphrasing attacks. Under back-translation attacks, SenDetEX further improves F1, AUC, and MCC by 7.1%, 5.8%, and 9.2%, respectively. Therefore, SenDetEX maintains strong robustness under various adversarial attacks.

We attribute this capability to two main factors: (1) perturbations that preserve semantic equivalence do not cause significant changes in the embeddings; and (2) our model considers and encodes contextual information, so modifying the input text does not affect the re-generated sentence and its inferred semantics.

4.3.4 Proxy Model and Sampling Strategy

We have already analyzed the context-aware length c in Section 4.3.1. In this section, we provide supplementary experimental results focusing on the impact of the sampling strategy (via the [MASK] rate) and the proxy model size (LLaMA-2 7B vs. 13B). We constructed two subsets from the XD dataset (detailed in Section 4.1): one with the highest proportion of LLM-generated sentences (donated as XD-max) and one with the lowest (donated as XD-min). The statistics of the supplementary datasets are shown in Table 7. The results in Table 8 show that SenDetEX consistently outperforms other baselines across both high and low [MASK] rate subsets, confirming its resilience to sampling strategy variation. Furthermore, larger proxy models (LLaMA-2 13B) yield slightly higher F1 scores, suggesting that stronger proxy models may lead to better detection, which is consistent with findings in (Mao et al., 2024).

4.3.5 Ablation Study

We conduct an ablation study on the XD subset, and the results are shown in Table 6. Compared to the vanilla SenDetEX, the variant SenDetEX-E shows a performance drop of 1.2%–1.4% in F1, 1.0%–

Source	Generator	Doc. count	[MASK] Rate (%)	Sen. Count (train/valid/test)	Subset Notification
XSUM	DeepSeek-V3	300	50.4	3185/1052/1099	XD-max
	DeepSeek-V3	300	15.3	3308/1086/1087	XD-min

Table 7: Statistics of the XD subsets. "Doc." and "Sen." denote "document" and "sentence", respectively. "[MASK] Rate" corresponds to the actual proportion of LLM-generated sentences.

Proxy Model	Method	XD-max	XD-min
LLaMA2-7B	SeqXGPT	91.3	90.8
LLaMA2-7B	POGER	93.0	92.5
LLaMA2-7B	SenDetEX	**98.0**	**97.0**
LLaMA2-13B	SeqXGPT	92.0	91.1
LLaMA2-13B	POGER	93.6	92.7
LLaMA2-13B	SenDetEX	**98.4**	**97.5**

Table 8: Detection results across different proxy models and methods on XD subsets with maximum (XD-max) and minimum (XD-min) LLM-generated sentence proportions under the in-domain scenario.

1.2% in AUC, and 1.7%–1.9% in MCC across all settings. This indicates that entropy information, as a complementary stylistic signal beyond probability, brings improvements to the overall detection performance. In the XD→XG, XD→WD, and paraphrasing settings, SenDetEX-R exhibits a drop of 3.4%–3.9% in F1, while SenDetEX-C shows a decrease of 4.4%–5.1%. These results demonstrate that contextual information significantly enhances performance under OOD and perturbation conditions, consistent with our attributions to its transferability and robustness in Sections 4.3.2 and 4.3.3. Furthermore, contextual and stylistic information complement SenDetEX's detection performance, validating the effectiveness of our joint modeling approach. We provide an extended analysis of the ablation study in Appendix G.

5 Conclusion

Sentence-level AI-generated text detection is both practically significant and technically challenging. Our work mainly addresses two major limitations of existing S-AGTD works: the lack of evaluation on complex human-AI hybrid content and the failure to incorporate contextual information. We first propose AutoFill-Refine, a high-quality synthesis method for human-AI hybrid texts, and construct a dedicated S-AGTD benchmark dataset. Inspired by cognitive science and preliminary experiments, we introduce the SenDetEX framework, which models and integrates stylistic and contextual information. Extensive experiments demonstrate the effectiveness of SenDetEX. We hope our work will provide valuable insights for future AGTD works.

Limitations

Although our proposed SenDetEX demonstrates promising performance, several limitations remain. First, SenDetEX's effectiveness relies on the preceding context of the sentence to be detected. Its advantage will diminish in sparse or noisy context scenarios, such as isolated sentences or fragmented documents. Second, the proxy model plays a crucial role in SenDetEX, and there is room for further exploration regarding the present fine-tuning strategy. Additionally, the prompts used in synthesizing our dataset can be further optimized. We plan to address these issues in future work.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant Nos. 72210107001, 72225011, 72434005, and L242400108, and by the CAS PIFI International Outstanding Team Project (2024PG0013).

References

Sara Abdali, Richard Anarfi, CJ Barberan, and Jia He. 2024. Decoding the ai pen: Techniques and challenges in detecting ai-generated text. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6428–6436.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. Fast-detectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In *The Twelfth International Conference on Learning Representations*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text*

with the natural language toolkit. " O'Reilly Media, Inc.".

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Joseph Cornelius, Oscar Lithgow-Serrano, Sandra Mitrović, Ljiljana Dolamic, and Fabio Rinaldi. 2024. Bust: Benchmark for the evaluation of detectors of llm-generated text. 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 8029–8057.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M 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.

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

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: Zero-shot detection of machine-generated text. *Preprint*, arXiv:2401.12070.

Abe Hou, Jingyu Zhang, Tianxing He, Yichen Wang, Yung-Sung Chuang, Hongwei Wang, Lingfeng Shen, Benjamin Van Durme, Daniel Khashabi, and Yulia Tsvetkov. 2024a. 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.

Abe Hou, Jingyu Zhang, Yichen Wang, Daniel Khashabi, and Tianxing He. 2024b. k-semstamp: A clustering-based semantic watermark for detection of machine-generated text. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1706–1715.

Andrew G Howard. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Guanhua Huang, Yuchen Zhang, Zhe Li, Yongjian You, Mingze Wang, and Zhouwang Yang. 2024. Are ai-generated text detectors robust to adversarial perturbations? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6005–6024.

Yifei Huang, Jiuxin Cao, Hanyu Luo, Xin Guan, and Bo Liu. 2025. Magret: Machine-generated text detection with rewritten texts. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8336–8346.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2. Minneapolis, Minnesota.

Ryuto Koike, Masahiro Kaneko, and Naoaki Okazaki. 2024. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21258–21266. Association for the Advancement of Artificial Intelligence (AAAI).

Md Tahmid Rahman Laskar, Sawsan Alqahtani, M Saiful Bari, Mizanur Rahman, Mohammad Abdullah Matin Khan, Haidar Khan, Israt Jahan, Amran Bhuiyan, Chee Wei Tan, Md Rizwan Parvez, Enamul Hoque, Shafiq Joty, and Jimmy Huang. 2024. A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13785–13816, Miami, Florida, USA. Association for Computational Linguistics.

Mina Lee, Percy Liang, and Qian Yang. 2022. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–19.

Alessandro Lenci et al. 2008. Distributional semantics in linguistic and cognitive research. *Italian journal of linguistics*, 20(1):1–31.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. 2024a. A semantic invariant robust watermark for large language models. In *ICLR*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024b.

Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Yepeng Liu and Yuheng Bu. 2024. Adaptive text watermark for large language models. In *International Conference on Machine Learning*, pages 30718–30737. PMLR.

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. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Yijian Lu, Aiwei Liu, Dianzhi Yu, Jingjing Li, and Irwin King. 2024. An entropy-based text watermarking detection method. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11724–11735.

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.

Chengzhi Mao, Carl Vondrick, Hao Wang, and Junfeng Yang. 2024. Raidar: generative AI detection via rewriting. In *The Twelfth International Conference on Learning Representations*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Diganta Misra, Trikay Nalamada, Ajay Uppili Arasanipalai, and Qibin Hou. 2021. Rotate to attend: Convolutional triplet attention module. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3139–3148.

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.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don' t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807.

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.

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.

Yuhui Shi, Qiang Sheng, Juan Cao, Hao Mi, Beizhe Hu, and Danding Wang. 2024. Ten words only still help: Improving black-box ai-generated text detection via proxy-guided efficient re-sampling. *arXiv preprint arXiv:2402.09199*.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Rafael Alberto Rivera Soto, Kailin Koch, Aleem Khan, Barry Y. Chen, Marcus Bishop, and Nicholas Andrews. 2024. Few-shot detection of machine-generated text using style representations. In *The Twelfth International Conference on Learning Representations*.

Zhixiong Su, Yichen Wang, Herun Wan, Zhaohan Zhang, and Minnan Luo. 2025. HACo-det: A study towards fine-grained machine-generated text detection under human-AI coauthoring. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22015–22036, Vienna, Austria. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Nafis Irtiza Tripto, Saranya Venkatraman, Dominik Macko, Robert Moro, Ivan Srba, Adaku Uchendu, Thai Le, and Dongwon Lee. 2024. A ship of theseus: Curious cases of paraphrasing in llm-generated texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6608–6625.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Lei Wang, Deke Guo, Huaming Wu, Keqiu Li, and Wei Yu. 2024a. Tc-gcn: Triple cross-attention and graph convolutional network for traffic forecasting. *Information Fusion*, 105:102229.

Pengyu Wang, Linyang Li, Ke Ren, Botian Jiang, Dong Zhang, and Xipeng Qiu. 2023. Seqxgpt: Sentence-level ai-generated text detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1144–1156.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, et al. 2024b. M4gt-bench: Evaluation benchmark for black-box machine-generated text detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3964–3992.

Junchao Wu, Runzhe Zhan, Derek F Wong, Shu Yang, Xuebo Liu, Lidia S Chao, and Min Zhang. 2025. Who wrote this? the key to zero-shot llm-generated text detection is gecscore. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10275–10292.

Yihuai Xu, Yongwei Wang, Yifei Bi, Huangsen Cao, Zhouhan Lin, Yu Zhao, and Fei Wu. 2025. Training-free LLM-generated text detection by mining token probability sequences. In *The Thirteenth International Conference on Learning Representations*.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Ruth Petzold, William Yang Wang, and Haifeng Chen. 2024. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text. In *ICLR*.

Shu Yin, Peican Zhu, Lianwei Wu, Chao Gao, and Zhen Wang. 2024. Gamc: an unsupervised method for fake news detection using graph autoencoder with masking. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, pages 347–355.

Xiao Yu, Yuang Qi, Kejiang Chen, Guoqiang Chen, Xi Yang, Pengyuan Zhu, Xiuwei Shang, Weiming Zhang, and Nenghai Yu. 2024. Dpic: Decoupling prompt and intrinsic characteristics for llm generated text detection. *Advances in Neural Information Processing Systems*, 37:16194–16212.

Cong Zeng, Shengkun Tang, Xianjun Yang, Yuanzhou Chen, Yiyou Sun, Zhiqiang Xu, Yao Li, Haifeng Chen, Wei Cheng, and Dongkuan DK Xu. 2024a. Dald: Improving logits-based detector without logits from black-box llms. *Advances in Neural Information Processing Systems*, 37:54947–54973.

Zijie Zeng, Shiqi Liu, Lele Sha, Zhuang Li, Kaixun Yang, Sannyuya Liu, Dragan Gašević, and Guanliang Chen. 2024b. Detecting ai-generated sentences in human-ai collaborative hybrid texts: Challenges, strategies, and insights. In *International Joint Conference on Artificial Intelligence, IJCAI 2024*, pages 7545–7553. International Joint Conferences on Artificial Intelligence.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatGPT interaction logs in the wild. In *The Twelfth International Conference on Learning Representations*.

Biru Zhu, Lifan Yuan, Ganqu Cui, Yangyi Chen, Chong Fu, Bingxiang He, Yangdong Deng, Zhiyuan Liu, Maosong Sun, and Ming Gu. 2023. Beat llms at their own game: Zero-shot llm-generated text detection via querying chatgpt. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7470–7483.

A Introduction of Baseline Methods

We categorize the baselines into training-free and supervised methods based on whether annotated training data is required. Unless otherwise specified, all methods follow the original experimental settings described in their respective papers.

A.1 Training-free Methods

Log Entropy (Gehrmann et al., 2019) assumes that AI-generated text exhibits lower "openness," i.e., the mean token entropy tends to be lower than that of HWT. We compute the mean log entropy of the detected text using the fine-tuned proxy model LLaMA2-7B (denoted as FTPM), consistent with the setup in Section 4.2.

Log Probability (Solaiman et al., 2019) assumes that AI-generated text has higher "precision," i.e., the mean token probability is typically higher than that of HWT. We use FTPM to compute the mean log probabilities of the detected text.

DNA-GPT (Yang et al., 2024) assumes that AGT differs from HWT in its N-gram distribution. DNA-GPT detects AGT by truncating the input and regenerating the missing portion using a re-generation model, then comparing the N-gram distributions between the original and re-generated text. We use FTPM as the re-generation model.

Fast-DetectGPT (Bao et al., 2024) is an enhanced version of DetectGPT (Mitchell et al., 2023), which posits that humans and AI exhibit discernible differences in token choice given a text. Fast-DetectGPT replaces the "perturbation" step in DetectGPT with a more efficient "sampling" process, improving detection efficiency and accuracy. We use FTPM to compute the token conditional probabilities.

A.2 Supervised Methods

SeqXGPT (Wang et al., 2023) is the first work designed for the S-AGTD task, which utilizes log probability lists from white-box LLMs as features for sentence-level detection. Its framework is based on convolutional and self-attention mechanisms. We use FTPM as the white-box LLM.

POGER (Shi et al., 2024) is a proxy-guided efficient re-sampling method, which selects a small subset of representative words to perform multiple rounds of re-sampling for AIGT detection. We use FTPM as the resampling model.

B Details of Fine-tuning Proxy Model

The work in (Zeng et al., 2024a) points out that fine-tuning the proxy model to align its probability distribution with the target model helps mitigate distribution shifts and enhances cross-model generalization. In our proposed SenDetEX framework (details in Section 3), the proxy model plays a crucial role—it is responsible for computing probability series, entropy series, self-embeddings, and contextual embeddings. Therefore, the fine-tuning process has the potential to further improve the detector's performance. Our fine-tuning configuration follows the Distribution-Aligned LLMs Detect (DALD) strategy proposed by (Zeng et al., 2024a). Specifically:

The fine-tuned model is LLaMA-2-7B, which serves as our base proxy model. In line with Section 4.1, the target detection models include GPT-4o and DeepSeek-v3 (temperature is set to 0.7). The training data is automatically generated via API calls to the target models, collecting 2,000 samples for each model. The data is structured as prompt-response pairs, denoted as $\mathcal{S} = \{(P_i, X_i)\}_{i=1}^{2000}$, where P_i is the prompt and X_i is the corresponding output from the target model. The prompts are sourced from the publicly shared prompts in the WildChat dataset (Zhao et al., 2024).

To enable parameter-efficient fine-tuning, we adopt the Low-Rank Adaptation (LoRA) technique (Hu et al., 2022), which introduces lightweight adapters while keeping the original model parameters frozen. The LoRA configuration is as follows: the rank is 16, and the scaling factor is 32. The adaptation modules are applied to several projection layers in LLaMA-2-7B, including `q_proj`, `v_proj`, `k_proj`, `o_proj`, `gate_proj`, `down_proj`, and `up_proj`.

The fine-tuning objective is to maximize the conditional likelihood of the target model outputs under the proxy model. Formally, the objective is:

$$\max_{\Theta} \sum_{[P,X]\in\mathcal{S}} \sum_{l=l(P)+1}^{l(P)+l(X)} \log p(y_l \mid y_{<l}; \text{sur} + \Theta)$$

where $l(X)$ denotes the length of the output text X, y_l is the l-th token to be predicted, and Θ represents the trainable parameters of the LoRA adapters. Gradients concerning the prompt P are blocked during training, and optimization is performed only on the generated part X.

During the fine-tuning process, the learning rate is set to 1e-4, with a per-device batch size of 1 and gradient accumulation steps set to 4 to enable multi-GPU parallelism. The maximum sequence length varies by model: 512 for GPT-4o and 2048 for DeepSeek-V3. The training is conducted on two NVIDIA A100 GPUs.

C Impact of Sentence Length on Detector Performance

In the experiment investigating the relationship between sentence length and the performance of AGTD methods, we randomly selected 1,000 documents from the WritingPrompt dataset (Fan et al., 2018), each containing both a "prompt" and a "long answer" field. We use DeepSeek-V3 and GPT-4o (in line with Section 4.1) as generators to produce AI-generated answers based on the "prompt" text, with the generation temperature set to 0.7. For each generator, we select 650 "long answer" documents as HWT and 350 AI-generated answers as AGT for evaluation. We make sure that each document contains at least 10 sentences.

We evaluate three training-free AGTD methods: Log Probability, Fast-DetectGPT, and DNA-GPT, whose detailed descriptions are provided in Appendix A. For each document under detection, we construct sub-documents by extracting the first N sentences, which serve as the input for the detector. The relationship between different values of N and the corresponding average F1 scores under each method is shown in Figure 2.

D Details of Adversarial Attacks

We evaluate two commonly used word-level attack strategies: random deletion and synonym substitution, where synonym substitution is implemented using the NLTK extension package (Bird et al.,

2009). with the character modification ratio fixed at 0.2.

In addition to word-level attacks, we conduct experiments in two sentence-level attacks: the paraphrasing attack and the back-translation attack. For the paraphrasing attack, we use the GPT-4o model (temperature is set to 0.7) with the following prompt design:

"Please rewrite the following sentence. The sentence is: [Insert your sentence here] The rewritten sentence is: "

For the back-translation attack, we use the GPT-4o model (temperature is set to 0.7) with the following prompt design:

"Please perform the following steps to finish the back-translation task: 1. Translate the following English sentence into German. 2. Then, translate the German version back into English. 3. Only return the translated English sentence. The sentence is: [Insert your English sentence here] The back-translation sentence is:".

According to the work of (Tripto et al., 2024) and others, for HWT, small-scale word-level modifications generally do not alter the attribution of authorship, which remains with the human. However, when the text undergoes large-scale modifications based on LLMs, the attribution of authorship may become ambiguous. Therefore, we generate four types of adversarial texts for each test sentence labeled as AI-based, using the aforementioned attack strategies. We only consider the random deletion and random substitution attacks for human-written sentences. We present two representative examples in Table 9.

E Special Cases of Re-generated Sentence

Our re-generated sentences are generated by the proxy model based on the preceding context of the sentence to be detected. The context-aware length c is a vital hyperparameter. No additional context is used when the target sentence s_i is at the beginning of the text ($i = 0$) or there is no context-aware information ($c = 0$). Following the work of (Mao et al., 2024), we use proxy model-generated synonymous rephrasings of the target sentence as approximations of the re-generated sentence. Specifically, we use the following prompt:

"Please rewrite the following sentence. The original sentence is: s_i. The rewritten sentence is: "

F Cases in Dataset Construction

In Table 11, we present two examples from the construction process of our human-AI hybrid benchmark dataset described in Section 4.1. The original texts in both cases are sourced from human-written summaries in the XSUM dataset, and the generator is DeepSeek-V3 (in line with Sections 4.1 and 4.2).

We also provide a detailed comparison between our dataset generated by AutoFill-Refine and four closely related datasets in CoAuthor(Lee et al., 2022), SeqXGPT(Wang et al., 2023), M4GT(Wang et al., 2024b), and HACo-Det(Su et al., 2025). The comparison is conducted across four dimensions:

- **Randomness**: Irregular alternation of human/AI sentences.
- **Label Clarity**: Only human or AI labels, without ambiguous collaborations.
- **Recency**: Incorporates recently developed LLMs (within the past three years).
- **Quality Control**: Applies filtering and cleaning strategies to ensure data quality.

As shown in Table 10, the dataset constructed using AutoFill-Refine simultaneously satisfies all these conditions and thus best meets the criteria for our S-AGTD tasks.

G Extended Ablation Analysis

To further illustrate the motivation of the SenDetEX design, we extend the ablation study beyond Section 4.3.5 and Table 6, providing a more systematic and progressive decomposition of the model architecture. The structure of our ablation settings is organized incrementally as follows:

- **(Setting 0) SeqXGPT:** Uses the probability series as input and adopts a "CNN + Transformer" architecture for feature extraction and classification.
- **(Setting 1) -E-C:** Disables all contextual information while retaining only the basic style signal. This setting is equivalent to Setting 0 in terms of input, and the performance gain here reflects the stronger temporal feature extraction capability of our Style Extraction Module.

Label	Human	AI
Original sentence	Underwater sonar equipment turned up a strange object more than two miles beneath the waves just before Christmas.	Honour crimes are a serious issue in the UK, with many cases going unreported due to fear and family pressure.
Random deletion	sonar equipment up a strange object than two miles beneath the waves just before Christmas.	Honour crimes are issue in UK, with many cases going unreported due to fear and family.
Random Substitution	Underwater sonar equipment turned up a foreign object more than ii miles beneath the waves just earlier Christmas .	Honour criminal offence are payoff in UK , with many cases going unreported due to fright and family.
Paraphrasing	(N.A.)	Honour crimes represent a significant problem in the UK, as numerous incidents remain unreported because of fear and pressure from families.
Back-translation	(N.A.)	Honour crimes are a serious problem in the UK, with many cases remaining unreported because of fear and pressure from the family.

Table 9: Two examples of adversarial attack cases on XSUM. "(N.A)": Not associated, meaning that for HWT, we do not consider its paraphrasing or back-translation scenarios.

Datasets	Randomness	Label Clarity	Recency	Quality Control
CoAuthor	Yes	No	No	No
SeqXGPT	No	Yes	Yes	No
M4GT	No	Yes	Yes	Yes
HACo-Det	Yes	Yes	Yes	No
Ours	Yes	Yes	Yes	Yes

Table 10: Comparison of datasets across four dimensions.

- **(Setting 2) -E-R:** Builds on Setting 1 by partially incorporating contextual information. The observed performance improvement demonstrates the value of intrinsic semantics.
- **(Setting 3) -C:** Builds on Setting 1 by leveraging the full style signal. The performance gain is attributed to the inclusion of openness information.
- **(Setting 4) -R:** Builds on Setting 1 by including both the full style signal and partial contextual information. This setting captures the synergistic effect between intrinsic semantics and openness.
- **(Setting 5) -E:** Builds on Setting 1 by incorporating full contextual information. Compared with Setting 2, the performance gain here demonstrates the added value of inferred semantics in combination with intrinsic semantics.

This progressive ablation strategy helps disentangle the contributions of each component and provides a more principled perspective on the fusion strategy in SenDetEX.

	Case 1	Case 2
Original document (d_i)	Home Office statistics show there were 22 arrests at Imps games between September 2014 and September 2015. [SEP] All were home fixtures, and 14 alone were made at a pre-season friendly against Doncaster Rovers. [SEP] Lincoln play in the National League - the fifth tier of English football. [SEP] Out of 12 banning orders issued in the same period, eight also came after the Doncaster game. $\cdots$	A leading activist, Nikolai Alexeyev, brought the case after the city authorities repeatedly rejected his requests to organise marches. [SEP] The Moscow authorities had argued the parades would cause a violent reaction. [SEP] But the court in Strasbourg said Russia had discriminated against Mr Alexeyev on the grounds of sexual orientation. [SEP] It said that by refusing to allow the parades, the authorities had "effectively approved of and supported groups who had called for (their) disruption". $\cdots$
Masked document (d_i')	Home Office statistics show there were 22 arrests at Imps games between September 2014 and September 2015. [SEP] [MASK] [SEP] Lincoln play in the National League - the fifth tier of English football. [SEP] Out of 12 banning orders issued in the same period, eight also came after the Doncaster game. $\cdots$	A leading activist, Nikolai Alexeyev, brought the case after the city authorities repeatedly rejected his requests to organise marches. [SEP] The Moscow authorities had argued the parades would cause a violent reaction. [SEP] [MASK] [SEP] It said that by refusing to allow the parades, the authorities had "effectively approved of and supported groups who had called for (their) disruption". $\cdots$
Shared prompt	Fill in each [MASK] in the following document with a single sentence to ensure overall fluency, coherence, and logic. Original document: d_i'. New completed document:	
Recovered document (d_i'')	Home Office statistics show there were 22 arrests at Imps games between September 2014 and September 2015. [SEP] **The majority of these arrests occurred during a match against Doncaster Rovers.** [SEP] Lincoln play in the National League - the fifth tier of English football. [SEP] Out of 12 banning orders issued in the same period, eight also came after the Doncaster game. $\cdots$	A leading activist, Nikolai Alexeyev, brought the case after the city authorities repeatedly rejected his requests to organise marches. [SEP] The Moscow authorities had argued the parades would cause a violent reaction. [SEP] **The European Court of Human Rights ruled in favor of Nikolai Alexeyev, stating that the Moscow authorities' ban on the marches was unjustified.** [SEP] It said that by refusing to allow the parades, the authorities had "effectively approved of and supported groups who had called for (their) disruption". $\cdots$
Perplexity Calculation	$\text{PPL}(d_i) = 26.83$ $\text{PPL}(d_i'') = 22.21$	$\text{PPL}(d_i) = 23.64$ $\text{PPL}(d_i'') = 24.88$
Decision	Accept	Reject

Table 11: Two examples from our benchmark dataset construction process through the AutoFill-Refine strategy, where [SEP] denotes the sentence separator. Bolded sentences indicate content generated by the LLM. From top to bottom, the table illustrates the steps of loading the original document, randomly masking the text, loading the shared LLM prompt, generating the recovered text, performing PPL verification, and making the final decision.