GL-CLiC: Global-Local Coherence and Lexical Complexity for Sentence-Level AI-Generated Text Detection

Rizky Adi[1], **Bassamtiano Renaufalgi Irnawan**[1], **Yoshimi Suzuki**[2], **Fumiyo Fukumoto**[2]
[1]Integrated Graduate School of Medicine, Engineering, and Agricultural Sciences
[2]Graduate Faculty of Interdisciplinary Research
{g24tka03,g23dtsa2,ysuzuki,fukumoto}@yamanashi.ac.jp
University of Yamanashi, Kofu, Japan

Abstract

Unlike document-level AI-generated text (AIGT) detection, sentence-level AIGT detection remains underexplored, despite its importance for addressing collaborative writing scenarios where humans modify AIGT suggestions on a sentence-by-sentence basis. Prior sentence-level detectors often neglect the valuable context surrounding the target sentence, which may contain crucial linguistic artifacts that indicate a potential change in authorship. We propose **GL-CLiC**, a novel technique that leverages both **G**lobal and **L**ocal signals of **C**oherence and **Lexi**cal **C**omplexity, which we operationalize through discourse analysis and CEFR-based vocabulary sophistication. **GL-CLiC** models local coherence and lexical complexity by examining a sentence's relationship with its neighbors or peers, complemented with its document-wide analysis. Our experimental results show that **GL-CLiC** achieves superior performance and better generalization across domains compared to existing methods.[1]

1 Introduction

The widespread adoption of large language models (LLMs) introduces significant social challenges, such as academic dishonesty and misinformation, indicating the need for robust AI-generated text detectors (Liang et al., 2025; Pudasaini et al., 2025). Although there is extensive research on document-level detection (Gui et al., 2025; Valdez-Valenzuela et al., 2025; Wang et al., 2024b; Verma et al., 2024; Yadagiri et al., 2024), these methods falter in real-world scenarios of collaborative human-AI writing, where documents are a mixture of human-written and machine-generated content (Yang et al., 2022; Dugan et al., 2023; Lee et al., 2022). This limitation highlights the need for AI-generated text (AIGT) detection at a finer granularity, as illustrated in Figure 1.

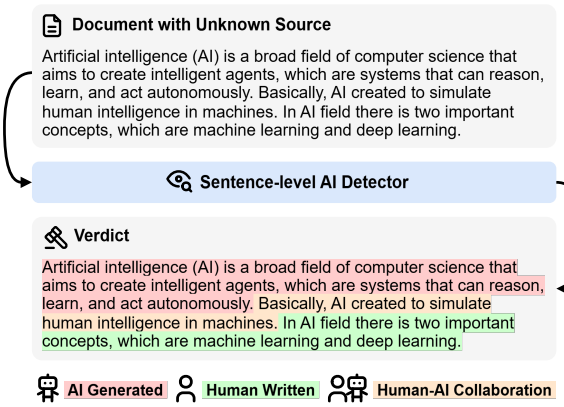

Figure 1: Sentence-level AIGT detection task in a human-AI collaborative writing scenario. The detector analyzes a document from an unknown source and assigns an authorship label (Human, AI, or Human-AI) to each sentence.

Although there has been previous work on sentence-level AIGT detection (Nguyen-Son et al., 2024; Zeng et al., 2024; Wang et al., 2023), current methods tend to focus on features of the target sentence alone (local features) (Zeng et al., 2024; Nguyen-Son et al., 2024) or the entire document without explicit sentence separation modeling (global features) (Wang et al., 2023). These approaches overlook the benefits of combining the global context with local details.

To address the limitations of current methods that rely solely on the global context of the document or local sentence details, we propose **GL-CLiC**, a sentence-level detector that incorporates linguistic signals from global and local perspectives. The core idea of **GL-CLiC** is to analyze two fundamental properties of text: coherence and lexical complexity from both local and global scopes. Specifically, our model analyzes local features by examining the narrative flow between adjacent sentences (local coherence) and the consistency of the lexical complexity of a sentence against its lexical group (local lexical complexity). This local analy-

[1]Our code is available at https://github.com/adirizq/gl-clic

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 3600–3617

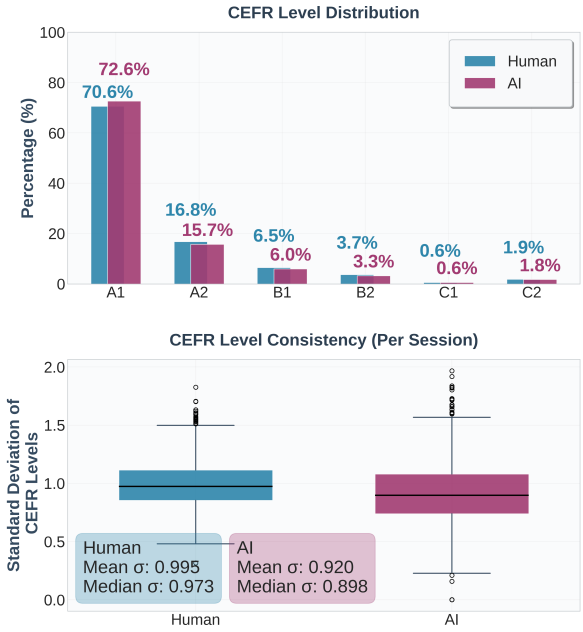

Figure 2: Analysis of CEFR level characteristics in the CoAuthor dataset. The top chart shows the distribution of CEFR levels across all words. The bottom chart shows a box plot of the standard deviation of CEFR levels within each writing session.

sis is complemented by a document-wide analysis, assessing the overall structural integrity of the text (global coherence) and its complete lexical profile (global lexical complexity). We operationalize these lexical features using the Common European Framework of Reference for Languages (CEFR), as it provides a robust proxy for vocabulary sophistication (Owen et al., 2021), allowing us to quantify the complexity of both local and global text spans.

Our focus on coherence and lexical complexity is motivated by the well-known linguistic differences between human and AI text. Previous work indicates that AIGT exhibits distinct coherence patterns (Liu et al., 2023; Sui et al., 2024) and often lacks the stylistic variation of human writing (Reinhart et al., 2025). Our CEFR level analysis on the CoAuthor dataset confirms that AI tends to generate text using simpler vocabulary and exhibits less stylistic variety than human writers. Our analysis, shown in Figure 2 (top), reveals that AI tends to use the A1-level vocabulary more frequently than humans. In contrast, humans use a broader range of more sophisticated vocabulary (A2-C2 levels). Furthermore, Figure 2 (bottom) also shows that AIGT exhibits a lower median standard deviation (SD) of CEFR levels, indicating a smaller lexical variation in AIGT. This suggests that AI generates text using a more consistent CEFR level, whereas human writing is more stylistically varied. This finding aligns with Reinhart et al. (2025), who report that AI prefers specific grammatical structures and struggles to replicate the stylistic diversity inherent in human text. Furthermore, by focusing on these basic linguistic features rather than topic-specific cues, **GL-CLiC** can learn more domain-agnostic authorship signals, improving its generalization across domains.

The main contributions of this paper can be summarized as follows:

(1) We propose **GL-CLiC**, a novel architecture that effectively integrates coherence and lexical complexity features at both local and global scopes for sentence-level AIGT detection, and

(2) We demonstrate that **GL-CLiC** consistently outperforms sentence-level AIGT detector baselines on both in-domain and cross-domain evaluation benchmarks.

2 Related Work

Sentence-Level AIGT Detection The growing trend of human-AI collaborative writing highlights the need for fine-grained detectors capable of operating at the sentence level (Zeng et al., 2024; Dugan et al., 2023; Lee et al., 2022). Existing methods can be broadly categorized by their focus on either global, document-level features or local, sentence-intrinsic features.

Early attempts at sentence-level AIGT detection relied solely on global features. For example, SeqXGPT (Wang et al., 2023) operates by modeling sequences of word log-probabilities from an open source LLM, treating the entire sequences as a signal for a transformer-based classifier, similar to speech processing. This reliance on log-probability features requires access to the source or a substitute LLM and incurs additional computational cost during feature generation. Furthermore, later work criticized the synthetic nature of its benchmark dataset (Zeng et al., 2024), advocating for more realistic data, e.g., the CoAuthor dataset (Lee et al., 2022). Crucially, SeqXGPT analyzes the document as a whole, without explicitly modeling sentence boundaries.

In contrast, newer attempts focus on local features. For instance, SimLLM (Nguyen-Son et al., 2024) measures the textual shift of a sentence after proofreading by LLM, hypothesizing that an AI-generated sentence will change less than human

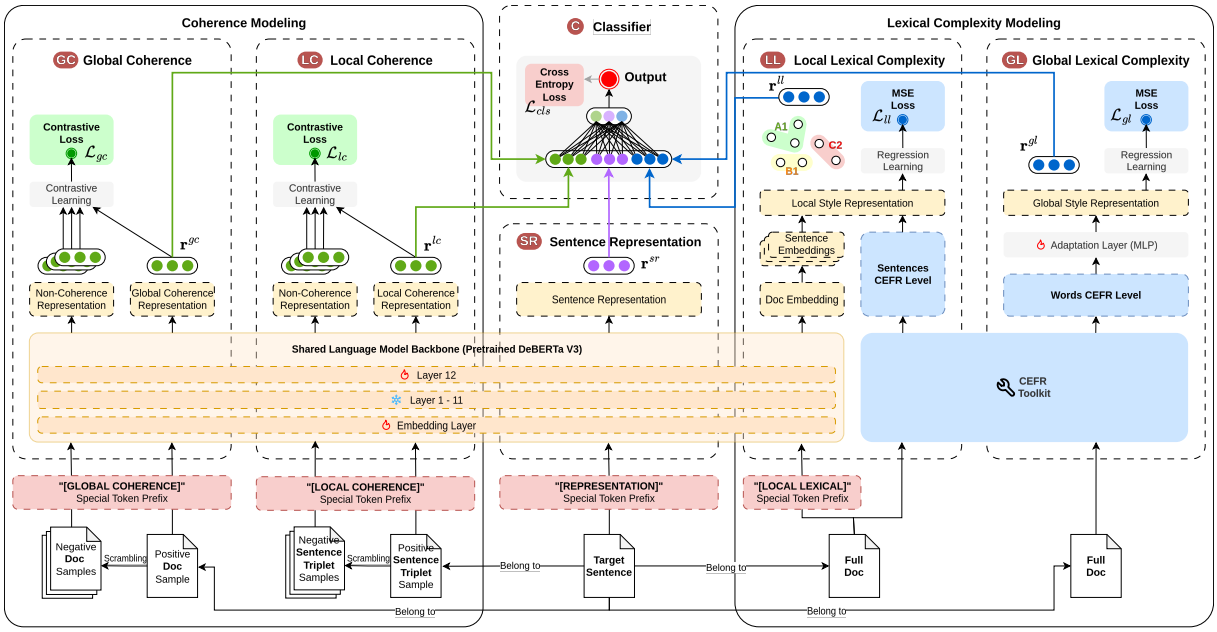

Figure 3: Overview of **GL-CLiC** framework, which consists of six modules: Global Coherence (GC), Local Coherence (LC), Global Lexical complexity (GL), Local Lexical complexity (LL), Sentence Representations (SR), and Classifier (C).

writing. However, this approach is computationally expensive and ignores the surrounding context, which is often vital for distinguishing human and AI writing. Recognizing the importance of context, Li et al. (2024) proposed the Paraphrased Text span Detection (PTD) framework, which analyzes the full text to assign a paraphrasing score to each sentence, demonstrating the value of surrounding context for this task. Despite these advances, we argue that existing approaches remain insufficient, as they either rely solely on isolated sentences, analyze the entire document without sentence separation, or depend on costly LLM inference. Instead, we propose a method that captures both local and global linguistic properties, such as coherence and lexical complexity.

Coherence Modeling Coherence modeling aims to assign a coherence score to an input text. Various works leverage contrastive learning to tackle this task (Cui et al., 2023; Jwalapuram et al., 2022) and the sentence disordering technique to produce the negative samples (Cui et al., 2023; Jwalapuram et al., 2022; Muangkammuen et al., 2020). Additionally, it is also known that AIGT exhibits a unique coherence pattern compared to human writing (Liu et al., 2023; Sui et al., 2024).

Lexical Complexity Modeling The analysis of lexical complexity, one of the components of stylometry (the study of writing style), has long been a focus of authorship attribution research that seeks to identify authors based on their unique linguistic habits (Sari et al., 2018; Kumarage et al., 2023). Such an analysis often examines lexical features such as word choice and vocabulary richness (Petukhova et al., 2024). Moreover, a recent study highlighted that LLMs exhibit a less varied style than humans, indicating that analyzing these lexical signals is a promising approach for AI detection tasks (Reinhart et al., 2025).

3 GL-CLiC Framework

The **GL-CLiC** framework integrates global and local features to distinguish AIGT, human-written, and human-AI collaboration sentences. As shown in Figure 3, **GL-CLiC** is built on a shared pretrained language model that serves as a backbone encoder. We design two parallel feature extraction modules that operate on the backbone outputs: coherence and lexical complexity modeling. The features produced by coherence and lexical complexity modeling are combined with direct sentence representation and fed into a final classification head, which is trained using a multi-task learning objective. Furthermore, we employ additional training techniques such as partial fine-tuning and the differential learning rate (DLR) training technique.

3.1 Task Definition

Let $D = (s_1, s_2, ..., s_n)$ be an input document composed of a sequence of n sentences. The task of detecting AI-generated text at the sentence level is to classify each sentence $s_i \in D$ into a label $\hat{l}_i$ from the predefined set $\mathcal{Y} = \{Human, AI, Human - AI\}$. The objective is to learn a model f that, for a given sentence s_i and its surrounding context D, predicts the corresponding label:

$$\hat{l}_i = f(s_i, D), \quad \text{where } \hat{l}_i \in \mathcal{Y}. \tag{1}$$

3.2 Coherence Modeling

Prior work observation shows that AIGT tends to have distinct coherence patterns (Liu et al., 2023; Sui et al., 2024). We hypothesize that these patterns are more pronounced as disruptions in textual coherence at both the document and sentence levels, particularly in texts with mixed authorship. Our **GL-CLiC** framework exploits this signal by incorporating a dedicated coherence component, illustrated on the left-hand side of Figure 3. This component comprises two parallel modules to learn the representation of global coherence ($\mathbf{r}_i^{gc}$) and local coherence ($\mathbf{r}_i^{lc}$).

GC: Global Coherence ($\mathbf{r}_i^{gc}$) The goal of this module is to capture potential disruptions in the document's overall narrative flow, which can be a strong artifact of mixed human/AI authorship. To model the global coherence feature, we follow the previous work (Jwalapuram et al., 2022; Cui et al., 2023) to define positive and negative samples. We use the original document as a positive sample and a sentence-shuffled version, which disrupts the narrative, as a negative sample. In addition, we mark the target sentence s_i by enclosing it with a special token `<hl>`. The final input format is $[s_1, s_2, ...,$`<hl>`s_i`<hl>`$, ..., s_{n-1}, s_n]$.

LC: Local Coherence ($\mathbf{r}_i^{lc}$) This module is designed to detect abrupt coherence breaks between adjacent sentences, which often signal a localized change in authorship. To capture the coherence between the target sentence and its neighboring sentence, we adapt the technique from Muangkammuen et al. (2020). The positive sample is the original sentence triplet (s_{i-1}, s_i, s_{i+1}). Negative samples are generated by taking the target sentence s_i and pairing it with random sentences "before" (s'_{i-1}) and "after" (s'_{i+1}) from elsewhere in the document. Similar to Global Coherence, the input is formatted as $[s_{i-1},$`<hl>`s_i`<hl>`$, s_{i+1}]$.

Contrastive Learning We train global and local coherence modules using a contrastive learning objective, which maps coherent sequences (positive) to similar representations in embedding space and pushes them far from incoherent sequences (negative). For both modules, the coherence representation ($\mathbf{r}_i^{gc}$, $\mathbf{r}_i^{lc}$) is the `[CLS]` token representation obtained from the shared backbone. We denote this generic representation as $\mathbf{r}_c$. These modules are trained to minimize a margin-based contrastive loss ($\mathcal{L}_{gc}$ and $\mathcal{L}_{lc}$) introduced by Jwalapuram et al. (2022). Both global and local coherence losses are defined as:

$$\mathcal{L}_c = -log\left(\frac{e^{f_\theta^c(\mathbf{r}_c^+)}}{e^{f_\theta^c(\mathbf{r}_c^+)} + \sum_{j=1}^{B} e^{(f_\theta^c(\mathbf{r}_{c_j}^-) - \mathcal{T})}}\right), \tag{2}$$

where f_θ^c is a linear projection that yields a coherence score, r_c^+ and $\mathrm{r}_{c_j}^-$ are the embeddings for the positive and j-th negative samples, respectively. B is the number of negative samples, and $\mathcal{T}$ is a margin hyperparameter.

3.3 Lexical Complexity Modeling

As mentioned in the Introduction, AI-generated text is often characterized by simpler, less varied word choices compared to human writing. Our lexical complexity module (right-hand side of Figure 3) is designed to explicitly quantify and capture these signals. In particular, we aim to capture the stylistic inconsistencies that arise in mixed-authorship documents by contrasting the document's global profile with the local style of a given sentence group. For example, if a simple sentence appears in an otherwise complex document, the final classifier can learn that the mismatch between local and global lexical complexity is a strong signal of mixed authorship. We use `cefrpy`[2] to extract the CEFR level of each word, assigning an integer level from 1 (A1) to 6 (C2). Based on these scores, we construct two lexical complexity features: a global document-level lexical complexity profile ($\mathbf{r}_i^{gl}$) and a group-based local lexical complexity representation ($\mathbf{r}_i^{ll}$).

GL: Global Lexical Complexity ($\mathbf{r}_i^{gl}$) This module computes the document-wide vocabulary profile. The intuition is that the global profile can help identify the simpler, less varied word choices often characteristic of AI text when compared to a typical human-written document. To represent the

[2] https://pypi.org/project/cefrpy/

overall lexical complexity of a document, we use the sequence of CEFR integer values of each word of document D_k that contains the target sentence s_i, which we denote as Fk. We either truncate or post-pad this sequence with negative one to a fixed length L. This fixed-length vector is then passed through a Multi-Layer Perceptron (MLP) p_ϕ to produce the global lexical complexity representation $\mathbf{r}_i^{gl} = p_\phi(F_k)$. This representation $\mathbf{r}_i^{gl}$ then fed into a linear projection f_θ^{gl} to compute the predicted global CEFR score $\hat{\bar{F}}_k = f_\theta^{gl}(\mathbf{r}_i^{gl})$. This module learns by aligning the predicted score $\hat{\bar{F}}_k$ with the actual average CEFR level of the document $\bar{F}_k$ through regression. The model learns by minimizing the mean squared error (MSE) loss function $\mathcal{L}_{gl}$ given by:

$$\mathcal{L}_{gl} = \frac{1}{N}\sum_{i=1}^{N}(\bar{F}_k - f_\theta^{gl}(p_\phi(F_k)))^2, \quad (3)$$

where N is the number of documents, F_k is a sequence of CEFR integer values for every word in the document k, $\bar{F}_k$ is the true average CEFR level, and $f_\theta^{gl}(p_\phi(F_k))$ is the model's predicted CEFR level for that document.

LL: Local Lexical Complexity ($\mathbf{r}_i^{ll}$) This module calculates a representation of a target sentence's "peer group" lexical complexity. The goal is to learn an embedding $\mathbf{r}_i^{ll}$ that captures the typical lexical complexity of sentences that are similar in style (as defined by CEFR level) to the target sentence s_i. This allows the final classifier to compare $\mathbf{r}_i^{ll}$ against the $\mathbf{r}_i^{gl}$ to spot inconsistencies. Given a target sentence s_i in a document, we first define its local peer group, G_m. We assume each sentence in the document has a pre-computed integer CEFR level, $C \in \{1, ..., 6\}$. The group G_m for s_i (which has level C_i) consists of all other sentences in the same document that also have the CEFR level C_i. We then obtain a contextualized embedding $\mathbf{e}$ for every sentence s in the group G_m. The local representation $\mathbf{r}_i^{ll}$ for the target sentence s_i is the mean-pooled embedding of all sentences in the group G_m. This representation is then fed to a linear projection layer f_θ^{ll} to compute the predicted G_m CEFR score $\hat{C}_i = f_\theta^{ll}(\mathbf{r}_i^{ll})$. This module learns through regression by aligning the predicted score $\hat{C}_i$ with the actual integer CEFR level C_i. This is achieved by minimizing the MSE loss function $\mathcal{L}_{ll}$:

$$\mathcal{L}_{ll} = \frac{1}{M}\sum_{i=1}^{M}(C_i - f_\theta^{ll}(\mathbf{r}_i^{ll})^2), \quad (4)$$

where M is the number of sentences, C_i is the true integer CEFR level for sentence s_i, and $f_\theta^{ll}(\mathbf{r}_i^{ll})$ is the predicted CEFR level for that sentence group.

3.4 Multi-Task Learning

The core idea of **GL-CLiC** is integrating these diverse signals of LC, GC, LL, and GL, which allows the model to spot contextual inconsistencies that a sentence-only model would miss. We implement this integration by jointly optimizing the main AIGT detection task and the four auxiliary tasks (GC, LC, GL, LL) using multi-task learning (MTL). For the main task, we concatenate the four auxiliary feature representations ($\mathbf{r}_i^{gc}$, $\mathbf{r}_i^{lc}$, $\mathbf{r}_i^{gl}$, $\mathbf{r}_i^{ll}$) with a direct sentence embedding $\mathbf{r}_i^{sr}$ (the `[CLS]` representation of s_i). This combined vector is fed into an MLP classifier, trained to predict the final label by minimizing a standard cross-entropy loss ($\mathcal{L}_{cls}$). All modules are optimized jointly via the final loss function:

$$\mathcal{L}_{final} = \mathcal{L}_{cls} + \alpha(\mathcal{L}_{gc} + \mathcal{L}_{lc} + \mathcal{L}_{gl} + \mathcal{L}_{ll}), \quad (5)$$

where α indicates a hyperparameter to control the influence of global coherence, local coherence, global lexical complexity, and local lexical complexity tasks, ensuring the main classification class remains as the central focus of the optimization process.

3.5 Additional Training Technique

Shared Backbone Model We used a shared backbone model in our global coherence, local coherence, local lexical complexity, and sentence representation modules. This shared model is a pre-trained language model (PLM), we specifically chose `DeBERTaV3` (He et al., 2023) based on findings by Zeng et al. (2024) that indicate its strong performance in distinguishing between human-written text and AIGT compared to other PLMs. Since this model is shared across multiple modules, we employ the task prefix technique introduced by Zhang et al. (2022) to facilitate effective multi-task learning.

Partial Fine-tuning To mitigate overfitting and prevent catastrophic forgetting of pre-trained knowledge, we only partially fine-tune the backbone model (Muñoz Sánchez et al., 2024). We freeze the first 11 transformer layers, allowing only the final layer and the embedding layer to be trained. The embedding layer should remain

	CoAuthor	SeqXGPT-Bench
Total Documents	1,445	30,000
Total Sentences	35,697	341,758
$Human$	23,183	129,453
AI	6,905	212,305
$Human - AI$	5,609	–

Table 1: Statistics of the datasets used in our experiments.

trainable to learn representations for our newly introduced task-prefix tokens.

Differential Learning Rate (DLR) We apply the differential learning rate technique introduced by Howard and Ruder (2018). In our experiments, we employ a lower learning rate η_{plm} for the PLM shared backbone model, and a higher learning rate η_{mlp} for the randomly initialized MLP components. This allows for slower fine-tuning of the already-trained PLM for knowledge preservation while enabling faster learning for the new layers being trained from scratch.

4 Experiments

4.1 Experimental Setup

Dataset We evaluate **GL-CLiC** on two sentence-level AI text detection benchmarks. Our primary dataset is **CoAuthor** (Lee et al., 2022), chosen, as it is, to our knowledge, the only dataset created through real human-AI collaborative writing interaction. It contains $1,445$ essays from 63 human writers, covering creative (830) and argumentative (615) writing settings. The dataset is originally labeled at the character level, so we follow the same data split and labeling technique as Zeng et al. (2024) to label the dataset into three classes: human, AI, and human-AI collaborative sentences. The second dataset, **SeqXGPT-Bench** (Wang et al., 2023) is used to test **GL-CLiC** generalization capabilities on a varied set of machine-generated texts. This dataset comprises $30,000$ documents synthesized using `GPT-2`, `GPT-J`, `GPT-Neo`, `Llama`, and `GPT-3.5-turbo`, each contributing $6,000$ documents. We adopt the original authors' sentence splitting and labeling procedure, resulting in a binary-labeled dataset, categorized as human or AI sentences. The statistics for both datasets are summarized in Table 1, while a detailed breakdown of the train, validation, and test splits is provided in Appendix A.

Evaluation Metrics Following Wang et al. (2023), we report per-class precision (P), recall (R), and F1 score, along with macro F1 for the overall performance.

Baselines We compare **GL-CLiC** against five strong baselines. These include: **SeqXGPT** (Wang et al., 2023), a feature-based method using LLM log-probabilities as input to a Transformer classifier; **PLM Fine-tuning** (Zeng et al., 2024), for which we use `DeBERTaV3-base` following its reported success on this task; **SimLLM** (Nguyen-Son et al., 2024), a recent approach that fine-tunes a PLM on a target sentence concatenated with its AI-proofread versions; **PTD** Framework (Li et al., 2024), which aims to detect AI paraphrased text span through sentence score regression; and **Prompted LLMs** (Labrak et al., 2024), where we perform zero-shot and few-shot classification using both `GPT-4o` and `Llama-4` (details in Appendix C).

Implementation We implement our model **GL-CLiC**, and all baselines using PyTorch and Hugging Face Transformers library. All experiments were conducted on a single NVIDIA RTX 6000 Ada GPU with 48GB of VRAM. To facilitate reproducibility, we provide comprehensive details on hyperparameter settings, optimizers, and training procedures in the Appendix D.

4.2 Main Results

Outperforming Baselines Table 2 shows the performance of our framework, **GL-CLiC**, compared against various baselines on the CoAuthor dataset. As shown in Table 2, our framework achieves a new state-of-the-art (SOTA) macro F1 score performance of 61.72, a 3.55 point increase from the second-best model.

Superiority over General-Purpose LLM The main result clearly shows the inadequacy of prompt-based LLM methods for this specialized task. Even a powerful GPT-4o model in a few-shot setting struggles, reaching only 36.79 of macro F1. This shows the need for specialized, dedicated detectors like **GL-CLiC**.

Effectiveness on Human-AI Class A key strength of **GL-CLiC** is its superior performance on the Human-AI collaboration sentence class, a category that proves particularly difficult for existing detection methods. It achieves the highest F1 scores for both AI and Human-AI classes. Notably, the performance on the Human-AI class marks a 18.23%

Method	Human			AI			Human-AI			Macro F1
	P	R	F1	P	R	F1	P	R	F1	
Zero-shot Llama 4	67.47	88.50	76.57	0	0	0	24.29	20.16	22.03	32.87
Zero-shot GPT-4o	67.83	93.04	78.46	23.33	0.66	1.28	29.18	14.40	19.28	33.01
Few-shot Llama 4	68.65	48.50	56.84	25.46	23.46	24.42	18.17	46.60	26.15	35.80
Few-shot GPT-4o	69.20	59.02	63.71	23.92	20.51	22.08	18.68	35.99	24.59	36.79
SeqXGPT	70.80	**95.97**	81.49	58.60	11.97	19.88	26.38	8.77	13.16	38.18
PLM Fine-tune	78.70	86.60	**82.46**	**58.68**	41.41	48.55	45.74	41.49	43.51	58.17
SimLLM	80.03	80.01	80.02	45.00	**52.61**	48.51	48.89	37.57	42.49	57.01
PTD	73.09	90.95	81.05	50.57	12.73	20.33	45.44	37.17	40.89	47.42
GL-CLiC Ours	**81.93**	82.59	82.26	54.08	49.10	**51.47**	**49.40**	**53.66**	**51.44**	**61.72**

Table 2: Sentence-level AI detection results on the CoAuthor Dataset. **Bold** marks the best performance and underline marks the second best.

Generator Model	Method	Human			AI			Macro F1
		P	R	F1	P	R	F1	
GPT-2	SeqXGPT	98.00	96.30	97.14	89.10	93.80	91.39	94.27
	PLM Fine-tune	79.34	75.09	77.15	92.40	93.93	93.16	85.16
	PTD	95.02	92.53	93.76	**97.68**	**98.48**	**98.08**	95.92
	GL-CLiC (Ours)	**98.99**	**97.69**	**98.34**	92.87	96.78	94.78	**96.56**
GPT-3.5-turbo	SeqXGPT	96.30	95.60	95.95	87.60	89.40	88.49	92.22
	PLM Fine-tune	97.05	81.06	88.34	93.82	99.15	96.41	92.37
	PTD	98.75	93.16	95.87	**97.76**	**99.61**	**98.68**	97.28
	GL-CLiC (Ours)	**99.42**	**99.06**	**99.24**	97.21	98.28	97.74	**98.49**
GPT-Neo	SeqXGPT	97.70	96.60	97.15	89.40	92.70	91.02	94.08
	PLM Fine-tune	81.06	79.09	80.06	93.49	94.20	93.85	86.95
	PTD	96.09	91.44	93.71	**97.45**	**98.87**	**98.16**	**95.93**
	GL-CLiC (Ours)	**98.32**	**97.67**	**97.99**	92.72	94.69	93.69	95.84
GPT-J	SeqXGPT	97.40	96.90	97.15	89.30	90.80	90.04	93.60
	PLM Fine-tune	77.93	70.06	73.78	92.41	94.84	93.61	83.70
	PTD	95.53	90.05	92.71	**97.08**	**98.74**	**97.91**	95.31
	GL-CLiC (Ours)	**98.80**	**98.18**	**98.49**	94.03	96.01	95.01	**96.75**
Llama	SeqXGPT	95.60	90.80	93.14	79.80	89.70	84.46	88.80
	PLM Fine-tune	70.33	73.60	71.93	88.94	87.24	88.08	80.01
	PTD	93.41	91.41	92.40	**96.50**	**97.35**	**96.92**	94.66
	GL-CLiC (Ours)	**97.58**	**96.55**	**97.06**	91.82	94.18	92.99	**95.02**

Table 3: Sentence-level AI detection results on the SeqXGPT-Bench Dataset. **Bold** marks the best performance and underline marks the second best within each generator model.

relative F1 score improvement over the next best baseline (PLM Fine-tune). This highlights that **GL-CLiC** architecture, which explicitly models coherence and lexical complexity patterns, is highly effective at capturing the subtle artifacts present in collaboratively generated text. Furthermore, **GL-CLiC** achieves the highest recall for the Human-AI class and remains highly competitive for the AI class, indicating its strong ability to retrieve sentences with any machine-generated content compared to other models. For a detailed analysis of the linguistic characteristics of Human-AI class, please see Appendix B.

4.3 Robustness Across Diverse Generators

To ensure our findings are not specific to the generator used in CoAuthor, we evaluated **GL-CLiC** on the SeqXGPT-Bench dataset, which contains texts from five different generator models. For these experiments, we trained and evaluated **GL-CLiC** and baselines independently for each generator model. As shown in Table 3, while the recent baseline (PTD) demonstrates strong results on the AI class, our **GL-CLiC** model consistently outperforms all baselines on the human class and maintains a competitive second-best on the AI class. This superior human class performance leads to a more balanced detection capability, ultimately achieving the highest Macro F1 score on 4 out of the 5 generator models and a very close second-best. These findings confirm that our method is not constrained to a single generator. By modeling fundamental signals of machine-generated text, such as coherence and lexical complexity, the **GL-CLiC** approach proves

Method	F1 Score			Macro F1
	H	AI	H-AI	
Creative → Argumentative				
SeqXGPT	78.72	2.17	14.53	31.81
PLM Fine-tune	78.79	43.26	42.11	54.72
SimLLM	76.71	42.38	32.41	50.50
PTD	77.23	0.75	23.98	33.99
GL-CLiC (Ours)	**79.51**	**46.31**	**57.37**	**61.06**
Argumentative → Creative				
SeqXGPT	**81.54**	9.48	4.32	31.78
PLM Fine-tune	73.67	42.62	30.60	48.96
SimLLM	69.77	39.36	26.69	45.27
PTD	77.58	0	27.15	34.91
GL-CLiC (Ours)	72.35	**43.77**	**33.47**	**49.86**

Table 4: Cross-domain experiments results, training on one domain and evaluating on another.

.

Method	F1 Macro			
	1-5	6-15	16-25	26+
SeqXGPT	39.41	36.01	37.39	36.91
PLM Fine-tune	41.48	50.21	58.86	55.89
SimLLM	27.45	27.98	**80.39**	47.90
PTD	35.69	42.83	45.19	46.23
GL-CLiC (Ours)	**51.97**	**58.60**	59.37	**56.58**

Table 5: Model performance across sentence lengths, from 1-5 word sentences to 26+ word sentences.

.

to be a robust and widely applicable framework for AI text detection.

4.4 Cross-Domain Generalization

To assess **GL-CLiC** generalization to unseen domains, we conducted a cross-domain experiment, which involved training in one text domain (e.g., Creative Writing) and evaluating in another domain (e.g., Argumentative Writing). As shown in Table 4, **GL-CLiC** demonstrates superior robustness by achieving the highest Macro F1 score in both *Creative → Argumentative* and *Argumentative → Creative* directions. One possible reason for that is **GL-CLiC** focuses on coherence and lexical complexity structure, allowing it to learn domain-agnostic signals. In contrast, baselines that may overfit topic-specific corpus cues experience a more significant performance degradation when the domain changes. Furthermore, **GL-CLiC** maintains a significant advantage in identifying the AI and Human-AI classes across domains, further validating our architectural design.

4.5 Short Sentence Performance

Our sentence length analysis in Table 5 confirms the robustness of **GL-CLiC**, which substantially outperforms all baselines on short (1-5 words) and medium-short (6-15 words) sentences. This highlights the strength of our local and global feature combination approach, which is ignored by other baselines. Additionally, the performance of SimLLM reveals a critical dependency on sentence length. Its "proofreading" comparison method fails on short sentences and overly complex long sentences (26+ words), but finds a "sweet spot" in the 16-25 word range, validating its underlying hypothesis only works within this specific bracket. This 16-25 word range also appears to be the performance peak for most methods, including PLM Fine-tune and **GL-CLiC**, which suggests this length may represent the majority of instances in the dataset. For a granular, per-class breakdown of **GL-CLiC** performance by sentence length, see Figures 6 and 7 in Appendix H.

Method	F1 Score			Macro F1
	H	AI	H-AI	
SeqXGPT	**81.47**	19.87	13.1	38.17
PLM Fine-tune	79.55	21.52	44.09	48.39
SimLLM	77.42	**30.14**	42.71	50.09
PTD	80.94	11.08	41.89	44.64
GL-CLiC (Ours)	77.89	27.00	**49.07**	**51.32**

Table 6: Paraphrase attack experiments results.

.

4.6 Paraphrase Attack

AIGT detectors are known to be vulnerable to paraphrase attacks (Wang et al., 2024a; Krishna et al., 2023). Following Nguyen-Son et al. (2024), we tested our method and other baselines' robustness using `GPT-3.5-Turbo` paraphrased AIGT test sentences (trained only on the original, unattacked training data). The results are shown in Table 6, which confirms that paraphrase attacks remain a significant threat to sentence-level detector models. The attacks significantly degrade the performance of models relying on rich stylistic features, especially in the AI class. This degradation is caused by the paraphrase process that modifies the lexical and syntactic clues these detectors rely on. Notably, the performance of SeqXGPT remains almost unchanged (38.18 Macro F1 in Table 2 vs 38.17 in Table 6). We hypothesize this is because its log-probability features were already insufficient for capturing these fine-grained artifacts, as evidenced by its low baseline F1 scores for the AI and Human-AI classes. Despite this susceptibility, our method still achieves the highest Macro F1 score, which

Method	F1 Score			Macro F1
	H	AI	H-AI	
Full Framework	82.26	51.47	**51.44**	**61.72**
w/o GC	84.48	49.69	49.26	61.15
w/o LC	**84.64**	40.22	49.02	57.96
w/o GL	84.40	**53.88**	45.95	61.41
w/o LL	84.47	49.40	42.68	58.85

Table 7: Ablation study results, through module removal.

indicates the most balanced performance under attack. However, we do acknowledge that **GL-CLiC** does not show the smallest relative performance drop, confirming that while it is comparatively robust, its features are still susceptible to this form of adversarial rewriting.

4.7 Ablation Study

Our ablation study, shown in Table 7, validates our integrated approach, as the full **GL-CLiC** framework achieves the highest Macro F1 score, and removing any single component leads to a net macro F1 performance loss. We also have the following observations:

(1) The results indicate that local features are important, as removing Local Coherence or Local Lexical Complexity causes the most significant macro F1 score drops (3.76 and 2.87 points, respectively). This highlights that closer context information is essential for sentence-level detection.

(2) The ablation study also reveals a complementary nature of feature combinations to the performance of each class. For instance, removing Global Lexical Complexity improves the F1 scores of Human and AI classes, but it comes at a steep cost to the detection of Human-AI class (a drop of 5.49 points). On the other hand, removing Local Coherence increases the performance of Human and Human-AI classes, but severely reduces AI class performance (a 11.25 point decrease).

(3) The result demonstrates that **GL-CLiC** is not just a collection of independent features, but a carefully balanced system. Its strength lies in integrating global, document-level signals with local, sentence-level cues to build a comprehensive text representation, allowing it to balance the performance on all classes, achieving the best macro F1 score.

For a detailed qualitative analysis of specific model predictions and learned feature representation analysis, see Appendix E and F. In addition to this module-level analysis, we provide a detailed ablation study on key hyperparameters, such as the auxiliary loss weight α, loss function, and the Differential Learning Rate (DLR) in Appendix G.

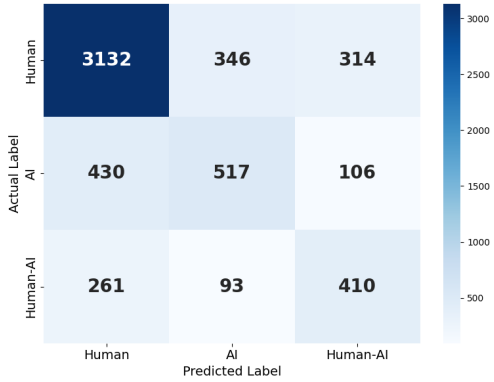

Figure 4: Confusion matrix of **GL-CLiC** predictions on the CoAuthor test set.

4.8 Error Analysis

We conducted an error analysis on the CoAuthor dataset, as this dataset is the main focus of this research. The analysis reveals a primary bias towards the Human class. The confusion matrix presented in Figure 4 shows that most misclassifications of the rare AI and Human-AI classes are predicted as Human. This suggests that when the model encounters ambiguous signals that do not strongly point to machine generation, it defaults to the high-frequency Human class seen during training. This bias is expected as the training data was unbalanced, with the majority of the data being Human sentences.

5 Conclusion

We presented **GL-CLiC**, a novel sentence-level AIGT detector that combines both global and local features of coherence and writing style. Our experiments showed that **GL-CLiC** outperforms existing baselines, proving especially effective for the human-AI collaboration category. Furthermore, we demonstrated the robustness of **GL-CLiC** across diverse generator models, sentence-length, and paraphrase attack. Our ablation study revealed that while the full framework provides the most balanced results, its strength lies in the complementary nature of its components, as removing features often improves one or two classes at the cost of others. Future work will explore other features and extend **GL-CLiC** to document-level detection.

Limitations

Domain and Language Generalization Our experiments are conducted exclusively on the CoAuthor dataset, which is composed of argumentative and creative essays in English. Consequently, the generalizability of **GL-CLiC** to other domains (e.g., scientific or medical text, news articles), genres (e.g., dialogue, social media), and languages other than English remains to be explored.

Generalization to Newer LLMs LLMs have improved rapidly since the collection of the CoAuthor dataset. The GPT-3 model, which was considered state-of-the-art at the time, has been surpassed by more powerful models. Consequently, the robustness of **GL-CLiC** has not yet been evaluated against text generated by newer models such as GPT-4o, Llama 4, or Google Gemini 2.5 family. These advanced models may produce text with greater human-like coherence and fewer grammatical artifacts, potentially posing a more significant challenge to our detection method.

Resilience to Adversarial Attack Our paraphrase attack experiments confirm that while **GL-CLiC** achieves the highest and most balanced Macro F1 score among the baselines, its features are still susceptible to this type of attack. This vulnerability is shown clearly by the significant performance degradation on the AI class, where the F1 score dropped from 51.47 to 27.00. Furthermore, this study does not explore the model's resilience against other common adversarial strategies, such as character-level perturbations, word-level substitutions, or more advanced style-obfuscation prompts.

Ethical considerations

This research follows the standards in NLP research. The data used in this study is only from publicly available sources, and personally identifiable information was not included.

Acknowledgments

We would like to thank anonymous reviewers and the metareviewer for their helpful comments and suggestions. This work is supported by the Support Center for Advanced Telecommunications Technology (SCAT) and JKA (2024M-557). Rizky Adi and Bassamtiano Renaufalgi Irnawan are funded by the MEXT scholarship (Grant Numbers. 244183 and 233203, respectively).

References

Jin Cui, Fumiyo Fukumoto, Xinfeng Wang, Yoshimi Suzuki, Jiyi Li, and Wanzeng Kong. 2023. Aspect-category enhanced learning with a neural coherence model for implicit sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11345–11358, Singapore. Association for Computational Linguistics.

Liam Dugan, Daphne Ippolito, Arun Kirubarajan, Sherry Shi, and Chris Callison-Burch. 2023. Real or fake text?: Investigating human ability to detect boundaries between human-written and machine-generated text. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):12763–12771.

Jiayi Gui, Baitong Cui, Xiaolian Guo, Ke Yu, and Xiaofei Wu. 2025. AIDER: a robust and topic-independent framework for detecting AI-generated text. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9299–9310, Abu Dhabi, UAE. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Prathyusha Jwalapuram, Shafiq Joty, and Xiang Lin. 2022. Rethinking self-supervision objectives for generalizable coherence modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6044–6059, Dublin, Ireland. Association for Computational Linguistics.

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

Tharindu Kumarage, Amrita Bhattacharjee, Djordje Padejski, Kristy Roschke, Dan Gillmor, Scott Ruston, Huan Liu, and Joshua Garland. 2023. J-guard: Journalism guided adversarially robust detection of AI-generated news. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–497, Nusa Dua, Bali. Association for Computational Linguistics.

Yanis Labrak, Mickael Rouvier, and Richard Dufour. 2024. A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical

and biomedical tasks. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2049–2066, Torino, Italia. ELRA and ICCL.

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*, CHI '22, New York, NY, USA. Association for Computing Machinery.

Yafu Li, Zhilin Wang, Leyang Cui, Wei Bi, Shuming Shi, and Yue Zhang. 2024. Spotting AI's touch: Identifying LLM-paraphrased spans in text. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7088–7107, Bangkok, Thailand. Association for Computational Linguistics.

Weixin Liang, Yaohui Zhang, Mihai Codreanu, Jiayu Wang, Hancheng Cao, and James Zou. 2025. The widespread adoption of large language model-assisted writing across society. *Preprint*, arXiv:2502.09747.

Xiaoming Liu, Zhaohan Zhang, Yichen Wang, Hang Pu, Yu Lan, and Chao Shen. 2023. CoCo: Coherence-enhanced machine-generated text detection under low resource with contrastive learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16167–16188, Singapore. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Meta. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation.

Panitan Muangkammuen, Sheng Xu, Fumiyo Fukumoto, Kanda Runapongsa Saikaew, and Jiyi Li. 2020. A neural local coherence analysis model for clarity text scoring. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2138–2143, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ricardo Muñoz Sánchez, David Alfter, Simon Dobnik, Maria Irena Szawerna, and Elena Volodina. 2024. Jingle BERT, jingle BERT, frozen all the way: Freezing layers to identify CEFR levels of second language learners using BERT. In *Proceedings of the 13th Workshop on Natural Language Processing for Computer Assisted Language Learning*, pages 137–152, Rennes, France. LiU Electronic Press.

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.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, and 400 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Nathaniel Owen, Prithvi Shrestha, and Stephen Bax. 2021. Researching lexical thresholds and lexical profiles across the common european framework of reference for languages (cefr) levels assessed in the aptis test. *ARAGs Research Reports Online*, ARG/2021(1).

Kseniia Petukhova, Roman Kazakov, and Ekaterina Kochmar. 2024. PetKaz at SemEval-2024 task 8: Can linguistics capture the specifics of LLM-generated text? In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1140–1147, Mexico City, Mexico. Association for Computational Linguistics.

Shushanta Pudasaini, Luis Miralles, David Lillis, and Marisa Llorens Salvador. 2025. Benchmarking AI text detection: Assessing detectors against new datasets, evasion tactics, and enhanced LLMs. In *Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect)*, pages 68–77, Abu Dhabi, UAE. International Conference on Computational Linguistics.

Alex Reinhart, Ben Markey, Michael Laudenbach, Kachatad Pantusen, Ronald Yurko, Gordon Weinberg, and David West Brown. 2025. Do llms write like humans? variation in grammatical and rhetorical styles. *Proceedings of the National Academy of Sciences*, 122(8).

Yunita Sari, Mark Stevenson, and Andreas Vlachos. 2018. Topic or style? exploring the most useful features for authorship attribution. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 343–353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Peiqi Sui, Eamon Duede, Sophie Wu, and Richard So. 2024. Confabulation: The surprising value of large language model hallucinations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14274–14284, Bangkok, Thailand. Association for Computational Linguistics.

Andric Valdez-Valenzuela, Helena Gómez-Adorno, and Manuel Montes-y Gómez. 2025. Text graph neural networks for detecting AI-generated content. In *Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect)*, pages 134–139, Abu Dhabi, UAE. International Conference on Computational Linguistics.

Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2024. Ghostbuster: Detecting text ghostwritten by large language models. 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 1702–1717, Mexico City, Mexico. Association for Computational Linguistics.

James Liyuan Wang, Ran Li, Junfeng Yang, and Chengzhi Mao. 2024a. RAFT: Realistic attacks to fool text detectors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16923–16936, Miami, Florida, USA. Association for Computational Linguistics.

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, Singapore. Association for Computational Linguistics.

Quan Wang, Licheng Zhang, Zikang Guo, and Zhendong Mao. 2024b. IDEATE: Detecting AI-generated text using internal and external factual structures. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8556–8568, Torino, Italia. ELRA and ICCL.

Annepaka Yadagiri, Lavanya Shree, Suraiya Parween, Anushka Raj, Shreya Maurya, and Partha Pakray. 2024. Detecting AI-generated text with pre-trained models using linguistic features. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 188–196, AU-KBC Research Centre, Chennai, India. NLP Association of India (NLPAI).

Daijin Yang, Yanpeng Zhou, Zhiyuan Zhang, Toby Jia-Jun Li, and Ray Lc. 2022. Ai as an active writer: Interaction strategies with generated text in human-ai collaborative fiction writing. In *Joint Proceedings of the ACM IUI Workshops*, volume 3124, pages 56–65. CEUR Workshop Proceedings.

Zijie Zeng, Shiqi Liu, Lele Sha, Zhuang Li, Kaixun Yang, Sannyuya Liu, Dragan Gasevic, and Guangliang Chen. 2024. Detecting ai-generated sentences in human-ai collaborative hybrid texts: Challenges, strategies, and insights. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 7545–7553. International Joint Conferences on Artificial Intelligence Organization. AI for Good.

Zhuosheng Zhang, Shuohang Wang, Yichong Xu, Yuwei Fang, Wenhao Yu, Yang Liu, Hai Zhao, Chenguang Zhu, and Michael Zeng. 2022. Task compass: Scaling multi-task pre-training with task prefix. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5671–5685, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Appendix

A Dataset Details

Split	Human	AI	Human-AI	Total
CoAuthor				
Train	15,969	4,855	3,490	2,8719
Val	3,422	997	883	6,247
Test	3,792	1,053	764	6,632
SeqXGPT-Bench				
Train	104,736	171,811	-	276,547
Val	11,315	19,208	-	30,523
Test	13,402	21,286	-	34,688

Table 8: Detailed statistics for the number of sentences in the train, validation, and test splits.

As mentioned in Section 4.1, we use the CoAuthor (Lee et al., 2022) and SeqXGPT-Bench (Wang et al., 2023) datasets. For CoAuthor, which is originally labeled at the character level, we follow the exact data split and sentence-level labeling methodology from Zeng et al. (2024) to create three classes (human, AI, human-AI). For SeqXGPT-Bench, we adopt the original authors' sentence splitting, labeling, and data splits, resulting in a binary-labeled (human, AI) dataset. Table 8 provides the detailed statistics for the training, validation, and test splits for both datasets.

B Human-AI Class Analysis

To better understand the "Human-AI" class, we analyzed 50 samples from the CoAuthor dataset. Our analysis reveals that:

(1) Human edits typically preserve the AI text core idea but add connectors, context, and details to improve flow.
(2) This process typically turns shorter, simpler AI sentences into longer, context-rich, more natural-sounding ones, increasing character length by ~60% on average.
(3) Human revisions do not consistently increase the text's CEFR level. Instead, the main objective is to enhance coherence.

These human edits cause PLM baselines to overpredict human-AI sentences as human. On the other hand, **GL-CLiC** remains more conservative when LL indicates a simpler lexical style relative to GL, increasing **GL-CLiC** performance on human-AI sentences.

Parameter	Value
Temperature	0.0
Top P	1.0
Top K	0.0
Frequency Penalty	0.0
Presence Penalty	0.0
Repetition Penalty	0.0
Min P	0.0
Top A	0.0

Table 9: LLM generation parameters. All parameters except Temperature are the OpenRouter default value.

Parameter	Value
Input	
Token Max Length	512
Batch Size	2
Learning Rate	
η_{plm}	0.00002
η_{mlp}	0.0001
Learning Rate Scheduler	
Factor	0.1
Patience	2 epochs
Monitor	Validation Loss
Interval	Epoch
Frequency	1
Early Stopping	
Mode	"max"
Patience	3 epochs
Monitor	Validation Macro F1-score
Training	
Max Epochs	10
Accelerator	GPU
Deterministic	True
α	1.0
Coherence Cost Function	
B	3
$\mathcal{T}$	0.1

Table 10: Training hyperparameters.

C LLM Classification Implementation

We use two leading models for our LLM experiments: the closed-source `GPT-4o` (OpenAI et al., 2024) and the open-source `Llama 4 Maverick` (Meta, 2025). Both model accessed via OpenRouter[3] using the model identifiers `"openai/gpt-4o"` and `"meta-llama/llama-4-maverick"`, respectively. The generation parameters for both models are shown in Table 9, while the Zero-shot and Few-shot prompts are presented in Figures 8 and 9.

Model	Version	Developer
ChatGPT GPT	3.5-turbo	OpenAI
GPT-4o	GPT-4o 2024-05-13	OpenAI
Yi	Yi 34B	01.AI
OpenChat	3.5 1210 7B	Alignment AI
Gemini	Gemini 1.5 Pro	Google
LLaMa	LLaMa 2 70B	Meta
Phi	Phi 2	Microsoft
Mixtral	8x7B Instruct v0.1	Mistral AI
QWen	QWen 1.5 72B	Alibaba
OLMO	7B Instruct	Allen AI
WizardLM	13B V1.2	WizardLM
Vicuna	13B v1.5	LMSYS

Table 11: LLM used as proofread generator on original SimLLM research (Nguyen-Son et al., 2024).

Model	Version	Provider
ChatGPT GPT	3.5-turbo	OpenRouter
GPT-4o	GPT-4o 2024-05-13	OpenRouter
Yi	Yi 34B Q8_0	ollama
OpenChat	3.5 1210 7B Q8_0	ollama
Gemini	Gemini 1.5 Pro	OpenRouter
LLaMa	LLaMa 4 Maverick	OpenRouter
Phi	Phi 2 Chat Q8_0	ollama
Mixtral	8x7B Instruct	OpenRouter
QWen	QWen 2.5 72B Instruct	OpenRouter
OLMO	7B Instruct hf Q8_0	ollama
WizardLM	13B V1.2 Q8_0	ollama
Vicuna	13B v1.5 Q8_0	ollama

Table 12: LLM used as proofread generator.

D Implementation Details

D.1 GL-CLiC Implementation Details

The **GL-CLiC** framework was optimized using AdamW (Loshchilov and Hutter, 2019), employing a differential learning rate scheme for the shared backbone PLM (η_{plm}) and randomly initialized MLP layers (η_{mlp}). The hyperparameter α was set to 1.0, which we found to provide the best performance after ablation study process. To ensure training stability and prevent overfitting, we utilized a `ReduceLROnPlateau` learning rate scheduler and implemented early stopping based on the validation macro F1-score. Our final model achieved a Macro F1 of 62.52 on the CoAuthor validation set, which is consistent with our test set performance and indicates no significant overfitting. The complete configuration of all hyperparameters is detailed in Table 10.

D.2 SimLLM Implementation Details

Originally, SimLLM uses 12 LLMs as a proofread generator, shown in Table 11. However, due to hardware limitations and restricted model access, we need to alter some of the original generator choices while choosing the most similar replacement. The LLM used as a proofread generator in our experiments can be seen in Table 12.

E Qualitative Analysis

To better illustrate how GL-CLiC integrated features help it succeed where a standard PLM fine-tuning baseline fails, we provide a qualitative analysis of predictions from the CoAuthor test set (Table 13). These cases demonstrate that the PLM baseline often fails due to a lack of information, as it is isolated to the target sentence only. It is important to note that these are plausible interpretations of which signals were supportive in these instances, not deterministic rules. The model learns the complex interplay between these features during training.

Case 1: AI text misclassified as Human	
Sentence	(Session 8462c...) "He's just here to help."
PLM Baseline	**Fails (Predicts Human):** Likely misclassified due to the short and simple, human-like phrasing.
GL-CLiC	**Correct (Predicts AI):** Leverages context. **SR** detects generic phrasing. The **LL** module identifies the sentence's simple style, while the **GL** module recognizes the document's complex vocabulary. This stylistic mismatch between the **LL** and **GL** strongly signals authorship change, pointing to AI generation (due to **LL** being simpler than **GL**).
Case 2: Human text misclassified as AI	
Sentence	(Session 5f43b...) "It is not our responsibility to keep up with what is going on in the world."
PLM Baseline	**Fails (Predicts AI):** Perhaps mistook the formal, argumentative tone for AI generation.
GL-CLiC	**Correct (Predicts Human): SR** detects argumentative phrasing. The **LC** module shows a smooth narrative flow with neighboring sentences, and the **LL** module finds the sentence's complex style matches the document's overall **GL** profile. This consistency, combined with the higher lexical complexity, strongly suggests human authorship.

Table 13: Qualitative analysis of GL-CLiC's predictions compared to a PLM baseline on misclassified examples from the CoAuthor test set.

[3] https://openrouter.ai

Figure 5: t-SNE Analysis of feature representation for each module of CoAuthor test set.

F Feature Analysis

To better understand the features learned by our auxiliary tasks, we visualized the feature representations from the CoAuthor test set using t-SNE (perplexity=50) and quantified their separability. We found that the classes form overlapping, non-convex structures, which is expected for sentence-level authorship analysis. Despite the visual overlap, the local neighborhoods are highly predictive. The 1-NN accuracy for all individual feature modules (GC, LC, GL, LL) significantly exceeded the 33% random baseline. Crucially, the final concatenated feature vector (used by the classifier) yielded the best local structure preservation (1-NN accuracy: 0.673). This supports our main claim and ablation study findings, which show the model's strength comes from integrating these complementary feature sets, as the combined representation is more predictive than any single component.

G Hyperparameter Ablation Study

Beyond the module-removal analysis, we conducted an additional ablation study on key hyperparameters. We focused on the auxiliary loss weight (α) from Equation 5, the Differential Learning Rate (DLR) technique, and alternative loss functions.

α	**Macro F1**
0.1	61.38
0.5	61.33
1.0	**61.72**

Table 14: Ablation study on the auxiliary loss weight α.

G.1 Auxiliary Loss Weight (α)

The α parameter balances the main classification loss ($\mathcal{L}_{cls}$) against the four auxiliary losses. As shown in Table 14, our ablation study revealed an interesting non-linear relationship between α and model performance, $\alpha = 0.5$ setting performed worse than both $\alpha = 0.1$ and $\alpha = 1.0$. We hypothesize that this non-linear result comes from the MTL optimization dynamics. The $\alpha = 0.5$ setting may create gradient conflicts between the main and auxiliary tasks, leading to a worse result. In contrast, the other settings provide clearer paths. $\alpha = 1.0$ (Strong Emphasis) prioritizes learning the beneficial auxiliary features, while $\alpha = 0.1$ (Subtle Guidance) uses them as an effective, non-conflicting regularizer for the main task. This suggests the "middle-ground" weight is suboptimal, and the auxiliary tasks are most effective as either a primary signal or a subtle regularizer.

Loss Function	Macro F1
Focal Loss	57.01
Weighted Cross-Entropy Loss	57.96
Cross-Entropy Loss	**60.84**

Table 15: Ablation study on the loss function.

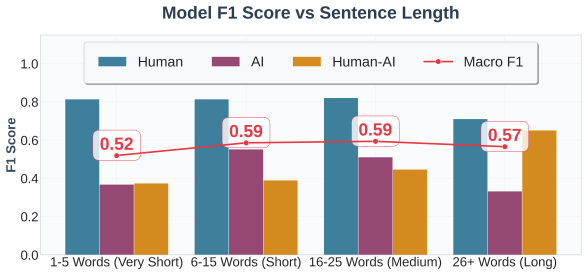

Figure 6: Impact of sentence length to F1 score performance.

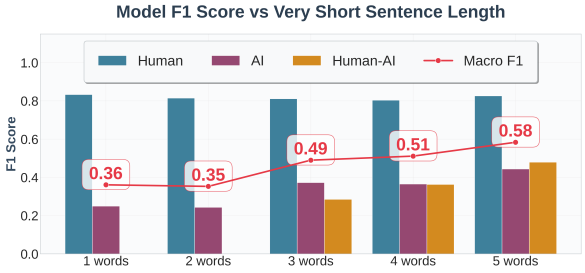

Figure 7: Impact of total words with F1 score performance on very short sentences.

G.2 Differential Learning Rate (DLR)

We removed DLR and used a single learning rate for all components, resulting in a significant 2.88 point decrease in Macro F1. This validates our hypothesis that DLR is a crucial component for stably fine-tuning the shared backbone while allowing the new, randomly initialized layers to learn quickly.

G.3 Alternative Loss Functions

In response to the data imbalance identified in the dataset, we experimented with class-balanced loss functions, including weighted focal loss and weighted cross-entropy. However, we found that this approach degraded performance (shown in Table 15), as it appeared to disrupt the carefully tuned balance of our multi-task learning objective. This finding justifies our use of the standard cross-entropy loss for the main classification task.

H Sentence Length Analysis

Figure 6 shows the performance of **GL-CLiC** tested against various sentence lengths, which indicates that **GL-CLiC** is weak against very short sentences (1-5 words). Figure 7 provides a more granular view, revealing a clear trend of performance deterioration with shorter sentences, with Macro F1 dropping from 0.58 (5-word sentences) to a mere 0.36 (1-word sentences).

```
You are an expert text classifier. Your task is to determine if the given sentence was written
purely by an AI, purely by a Human, or through Human-AI collaboration.

Carefully examine the sentence below. Consider its style, complexity, tone, and potential
signs of editing or integration.

* Use "AI" if the sentence appears entirely generated by an AI.
* Use "Human" if the sentence appears entirely written by a human.
* Use "AI-Human" if the sentence shows characteristics of both AI generation and human
writing/editing (e.g., AI text modified by a human, human text with AI-generated parts, or
a blend of styles).

Your response must be *exactly* one of the following labels: "AI", "Human", or "AI-Human".
No other text or explanation is allowed.

Sentence to Classify:
[sentence]

Verdict:
```

Figure 8: Zero-shot prompt for LLM inference. The "**[sentence]**" is changed to the target sentence during inference.

```
You are an expert text classifier. Your task is to determine if the given sentence was written
purely by an AI, purely by a Human, or through Human-AI collaboration.

Carefully examine the sentence below. Consider its style, complexity, tone, and potential
signs of editing or integration.

* Use "AI" if the sentence appears entirely generated by an AI.
* Use "Human" if the sentence appears entirely written by a human.
* Use "AI-Human" if the sentence shows characteristics of both AI generation and human
writing/editing (e.g., AI text modified by a human, human text with AI-generated parts, or
a blend of styles).

Your response must be *exactly* one of the following labels: "AI", "Human", or "AI-Human".
No other text or explanation is allowed.

- - -
**Examples:**

Sentence to Classify:
He's learning that he doesn't need to change his appearance, but he does need to start
changing his behavior.
Verdict:
AI

Sentence to Classify:
Instead he decides to spend that time at home learning to cook a new recipe.
Verdict:
Human

Sentence to Classify:
Matt is a good sport about it all, and even helps Will Smith with his investigation.
Verdict:
AI-Human

Sentence to Classify:
he wolf totally didn't know what to make of this house.
Verdict:
AI

Sentence to Classify:
That's exactly why Donald Trump did not lose by a landslide.
Verdict:
Human

Sentence to Classify:
One walked forward, and to his surprise, began speaking in English.
Verdict:
AI-Human
- - -

**Now, classify the following sentence:**

Sentence to Classify:
[sentence]

Verdict:
```

Figure 9: Few-shot prompt for LLM inference. The examples provided in the prompt come from the training set. The "**[sentence]**" is changed to the target sentence during inference.