AEyeDE: An Attention-Based Attribution Framework for AI-Generated Text Detection

Anonymous ACL submission

Abstract

Detecting AI-generated text is becoming increasingly challenging as modern language models approach human-level fluency and can evade detectors that rely on surface statistics or likelihood-based signals. We propose AEyeED an attribution-driven approach to human-AI authorship detection that leverages model attention as a discriminative signal. Specifically, we extract attention-based attribution matrices for both human- and AI-generated text using a *proxy* Transformer model with white-box access and train a lightweight Convolutional Neural Network to learn representations from these attribution maps. Across standard benchmarks, our method achieves performance competitive with state-of-the-art detectors using both encoder-decoder machine translation and decoder-only open-ended generation settings. We further provide evidence that attention maps exhibit detectable recurring local structures whose relative frequency differs reliably between human and AI text across datasets and proxy models. We will make the code publicly available for future research.

1 Introduction

The advent of large language models (LLMs) has enabled the generation of coherent, context-aware, and human-like text across a wide range of domains and languages (Naveed et al., 2023; Chang et al., 2024). While these advances unlock substantial benefits, they also raise critical challenges related to information integrity, authorship, and misuse, including large-scale misinformation in journalism and academic dishonesty in educational settings, where automated content generation threatens societal trust, originality, and assessment validity (Dugan et al., 2023; Wu et al., 2025; Liu et al., 2024b; Ali et al., 2025; Huang et al., 2025b; Bittle and El-Gayar, 2025).

In response, several AI-generated text detection methods has been proposed, including surface-statistical approaches exploiting cues such as perplexity, burstiness, and n-gram repetition (Gehrmann et al., 2019; Ippolito et al., 2020); likelihood-based methods that probe instability in a model's probability landscape via perturbation or re-sampling (Mitchell et al., 2023; Bao et al., 2023); supervised classifiers that fine-tune Transformer encoders on labeled data (Li et al., 2025; Zhu et al., 2025); and watermarking techniques that embed detectable signals for source attribution (Kirchenbauer et al., 2023; Liu et al., 2024a). However, each paradigm has intrinsic limitations: statistical and likelihood-based detectors degrade as LLMs are optimized to mimic human distributions through techniques such as RLHF (Christiano et al., 2017); supervised classifiers suffer under domain shift and unseen generators (Uchendu et al., 2021); and watermarking requires model-side cooperation and is fragile to paraphrasing, post-editing, or partial reuse (Liu et al., 2024a; Wang et al., 2025; Niess and Kern, 2025; Ahn et al., 2025). As a result, robust detection remains an open challenge, continually undermined by advances in generation quality and adversarial evasion strategies (Wu et al., 2025, 2024).

These challenges motivate a shift away from detecting *what* is written toward analyzing *how* text is produced. Existing detection methods largely rely on surface statistics, likelihood signals, or model-dependent mechanisms such as watermarks, all of which can be sensitive to distribution alignment, paraphrasing, or model evolution. We argue that robust and generalizable detection should instead exploit internal behavioral traces of neural language models. In particular, we hypothesize that the attribution patterns induced during text processing differ structurally between LLM-generated and human-authored text, and that this information can be used to distinguish between them.

To test this hypothesis, we introduce **AEyeDE**, an attribution-based detection framework that op-

erates directly on attention-based attribution maps extracted from Transformer models (Vaswani et al., 2017). Given an observed text x (human or AI), AEYEDE passes x through a fixed *proxy* model G_θ with white-box access and derives an attention-based attribution matrix (Sec. 2); this provides structured evidence that complements raw-text cues. We process attribution maps using a multi-scale convolutional encoder with attention pooling to obtain compact embeddings for authorship classification (Figure 1), making the detector less sensitive to purely lexical or stylistic variation and enabling evaluation under generator shift.

Beyond detection accuracy, we analyze what the CNN attribution encoder captures in attention maps. Clustering 8×8 patches in its last convolution stage feature space reveals recurring local patterns (*motifs*) whose prevalence differs between human and AI-generated text across datasets and proxy models. This indicates that authorship leaves a localized, repeatable signature in proxy-model attention maps that our detector can exploit.

We validate AEYEDE across both encoder–decoder and decoder-only paradigms: machine translation task (WMT14 and the UN Parallel Corpus) and open-ended generation datasets (HC3 and RAID), covering multiple languages, domains, and model families under realistic prompt-free evaluation settings.

Our main contributions are summarized as follows:

- We introduce **AEyeDE**, an attribution-conditioned framework that uses attention-based attribution maps from a proxy Transformer as structured evidence for AI-generated text detection.
- We show that modeling attribution structure with a lightweight CNN yields a robust detection signal that generalizes across encoder–decoder and decoder-only models, multiple datasets, and LLM families.
- We find that the attention maps of the proxy model contain distinct local patterns, "motifs", whose share systematically differs between human and AI generation, providing an interpretable, localized signature of authorship even when motif prevalence is only weakly aligned with saliency/ablation scores.

2 AEyeDE Framework

We formalize AI-text detection as a binary classification problem in which the detector has white-box access to a *proxy* language model G_θ, chosen to be either (i) the suspected generator or (ii) a capable surrogate model. Given an observed text x, we pass x through G_θ and extract attention weights, averaged across heads across layers, to obtain an attention-based attribution map. More specifically, we compute these attributions using the same proxy model for all inputs, regardless of whether x is human-written or model-generated. Our hypothesis is that, even when surface-level statistics may already differ between sources (Bao et al., 2023), the internal dynamics of G_θ when processing x induce systematic and detectable differences in the resulting attention maps for human versus AI text.

The resulting attention attributions can be viewed as weight matrices that quantify token-to-token influence during generation. In the encoder–decoder (machine translation) setting, the attributions describe the influence of *source* tokens on each *target* token (cross-attention). In the decoder-only setting, the attributions describe the influence of *previous* tokens on the next-token predictions (causal self-attention). In both cases, the attribution map serves as the primary input to our downstream detector.

Let $\mathcal{V}$ be a vocabulary and let $x = (x_1, \ldots, x_T) \in \mathcal{V}^T$ denote a text sequence of length T. Accordingly, AI-text detection as a binary classification with a label

$$\begin{aligned} &y \in \{0, 1\}, \\ &y = 1 \text{ (AI-generated)}, \\ &y = 0 \text{ (human-written)}. \end{aligned}$$

Having the candidate generator LLM G_θ (a Transformer with parameters θ) that is *supposed* to have produced x, the detector is a conditional classifier:

$$D_\phi(\, x; \theta\,) \;=\; f_\phi\big(x,\; \psi(x; \theta)\big) \in [0, 1],$$

where $\psi(x; \theta)$ is a feature representation derived from G_θ (here: attention-based attributions) and f_ϕ is a learned decision function with parameters ϕ.

Assume G_θ is an L-layer, H-head causal Transformer. At layer $\ell \in \{1, \ldots, L\}$ and head $h \in \{1, \ldots, H\}$, self-attention produces query/key matrices $Q^{(\ell,h)}, K^{(\ell,h)} \in \mathbb{R}^{T \times d_k}$. The (masked) atten-

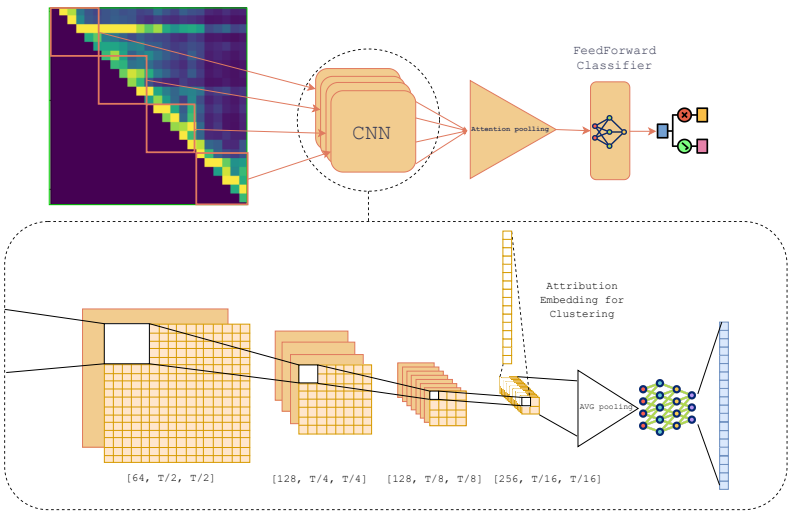


Figure 1: Overview of the proposed attribution-based detector. Given a text sample and access to a proxy generator model G_θ, we extract an attention-based attribution matrix A (top-left). We summarize A by sampling fixed-size square blocks (e.g., 128×128) along the main diagonal that captures the richest local token–token interactions (Highlighted by orange frames inside the heatmap) (Xiao et al., 2023; Ivanitskiy et al.; Qi et al., 2025). Each block is encoded by a CNN (bottom; initial convolutional stage layer + four convolutional stages with 2×2 max-pooling, followed by an intermediate feedforward network), producing a representative embedding per block. We aggregate these embeddings with learnable attention pooling to form a global attention map representation, which is fed to a lightweight feedforward classifier to predict human vs. LLM authorship (top-right). The 16×16 feature map after the last convolution stage in CNN is associated with patches (white squares) of original attention attribution for clustering.

tion attribution matrix is

$$A^{(\ell,h)}(x;\theta) = \text{softmax}\left(\frac{Q^{(\ell,h)}(K^{(\ell,h)})^\top}{\sqrt{d_k}} + M\right) \in \mathbb{R}^{(T+1)\times T}(\text{for brevity}\mathbb{R}^{(T)\times T}).$$

where $M \in \mathbb{R}^{(T+1)\times T}$ is the causal mask starting from <bos> token , Thus, each entry

$$a_{t,s}^{(\ell,h)} = \left[A^{(\ell,h)}(x;\theta)\right]_{t,s}$$

quantifies the influence (importance weight) of token position s on the representation used to predict token position t averaged across all layers and attention heads. We use the Inseq Python library to derive these attention attributions (Sarti et al., 2023). The notation is generalizable to the encoder-decoder model, where $s \neq t$.

The architecture of the main attention-based detector depicted in 1 consists of a CNN that takes patches of information from the main diagonal of the attribution matrix for the decoder-only models and the whole matrix for the encoder-decoder model[1]. Given a target text sequence $y_{1:T}$ and an attribution map A extracted from a candidate LLM, We build a classifier that can optionally use (i) a *text* representation of $y_{1:T}$ and (ii) an *attribution* representation computed from A. For readability, we describe the per-example computation and omit the batch dimension.

Let G_θ denote the candidate LLM. We obtain an attribution map $A \in \mathbb{R}^{T_x \times T_y}$ from G_θ (e.g., an attention-derived attribution matrix), where rows and columns index source/previous tokens and target/current tokens, respectively.

We summarize the 2D attribution map by extracting square blocks of size w_a along a diagonal traversal. We set the attribution block size to $w_a = 128$, i.e., we extract square blocks

$$A_k \in \mathbb{R}^{128\times 128}.$$

The reason is that we assume the most information in the decoder-only models to be located around the main diagonal (Xiao et al., 2023; Ivanitskiy et al.; Qi et al., 2025). Let (p_k, q_k) denote the top-left corner of the k-th block, and define

$$A_k = A\big[p_k : p_k + w_a,\ q_k : q_k + w_a\big]. \quad (1)$$

We mark a block as valid if it contains at least one

[1] In our experiments with the encoder-decoder model, we used samples of at most 128 tokens. For this model, we used just one patch that covers the whole attribution matrix.

non-padding entry:

$$u_k \;=\; \mathbb{I}\Big(\sum\nolimits_{i,j} M\big[p_k+i,\; q_k+j\big] > 0\Big) \in \{0,1\}. \tag{2}$$

Each block is encoded by a CNN-based attribution encoder

$$b_k \;=\; E_{\text{attr}}(A_k) \in \mathbb{R}^{d_{\text{attr}}}. \tag{3}$$

The CNN-based attribution encoder E_{attr} treats A_k as a single-channel image and applies five convolutional stages with 2×2 max-pooling, followed by global average pooling and an MLP:

$$b_k \;=\; W_2\,\rho\big(\mathrm{BN}(W_1\,g(A_k)+\beta_1)\big)+\beta_2, \tag{4}$$

$\rho(\cdot)$ represents ReLU and $g(\cdot)$ denotes the convolutional pipeline with channel progression $1 \to 32 \to 64 \to 128 \to 128 \to 256$. After the last convolution stage a feature map transforms to $16 \times 16 \times 256$, which, after global average pooling, is reduced to a single embedding $b_k \in \mathbb{R}^{d_{\text{attr}}}$.

Optionally, we may process the text as complementary information[2] We assume padding masks $m_x \in \{0,1\}^{T_x}$ and $m_y \in \{0,1\}^{T_y}$ are available, and define

$$M \;=\; m_x\, m_y^{\top} \in \{0,1\}^{T_x \times T_y}, \tag{5}$$

which marks valid (non-padding) attribution entries. To handle long sequences, we partition the (padded) token sequence into fixed-length windows of size w_t. Let $N = T/w_t$ and define chunks

$$y^{(i)} \;=\; y_{(i-1)w_t+1:\,iw_t}, \qquad i = 1,\ldots,N. \tag{6}$$

A text encoder E_{text} maps each chunk to a vector

$$c_i \;=\; E_{\text{text}}\big(y^{(i)}\big) \in \mathbb{R}^{d_{\text{text}}}. \tag{7}$$

We ignore fully padded chunks via a validity indicator $v_i \in \{0,1\}$. Next, we aggregate a set of vectors $\{z_i\}_{i=1}^{n}$ (either $\{c_i\}$ or $\{b_k\}$) using learnable attention pooling. Given validity mask $m_i \in \{0,1\}$, we compute

$$s_i \;=\; a^{\top}\tanh(W z_i), \tag{8}$$

$$\alpha_i \;=\; \frac{m_i\,\exp(s_i)}{\sum_j m_j\,\exp(s_j)}, \tag{9}$$

$$\mathrm{Pool}(\{z_i\},\{m_i\}) \;=\; \sum\nolimits_i \alpha_i z_i. \tag{10}$$

[2]We evaluate this text-augmented variant for encoder–decoder models. In our experiments, attention-only representations are consistently competitive with the text-augmented setting; therefore, we emphasize the attention-based results in the main paper.

This yields an optional text summary h_{text} from $\{c_i\}$ and an attribution summary h_{attr} from $\{b_k\}$:

$$h_{\text{text}} = \mathrm{Pool}(\{c_i\},\{v_i\}), h_{\text{attr}} = \mathrm{Pool}(\{b_k\},\{u_k\}). \tag{11}$$

Finally, we form a fused representation by concatenating the available components:

$$h \;=\; [\,h_{\text{text}};\, h_{\text{attr}}\,]. \tag{12}$$

A two-layer MLP produces a scalar logit ℓ and a probability via the sigmoid:

$$\ell \;=\; w^{\top}\rho(W_h h + b_h) + b, \tag{13}$$

$$p(y{=}1 \mid y_{1:T}, A) \;=\; \sigma(\ell). \tag{14}$$

The model is trained using binary cross-entropy on labeled examples.

One should note that discarding prompts in the decoder-only setting corresponds to conditioning the detector only on the observed output text (and its derived attributions), i.e., on $(y_{1:T}, A)$ rather than on the prompt–response pair.

3 Experimental Results

3.1 Datasets

In this study, we evaluate both encoder–decoder and decoder-only language models. For the encoder–decoder setting, we conduct experiments on three translation language pairs: French–English (fr-en) and German–English (de-en) using WMT14 (Bojar et al., 2014), and Arabic–English (ar-en) using the UN Parallel Corpus (Ziemski et al., 2016). For each language pair, we construct a dataset of 200k source–target examples consisting of gold (human) reference translations and corresponding model-generated translations produced by Marian-MT model (Tiedemann and Thottingal, 2020). We selected samples with source and target pairs of at most 128 tokens.

For the decoder-only setting, we use the HC3 dataset (Guo et al., 2023) and the RAID corpus (Dugan et al., 2024). HC3 provides paired human and ChatGPT (OpenAI, 2023) responses. We remove examples exceeding 1,024 tokens and retain approximately 24k samples per class. RAID contains human-written text and model-generated text spanning multiple domains. From RAID, we use outputs from Cohere, LLaMA 2 70B (chat), GPT-2 XL, GPT-3, and Mistral-7B, together with the corresponding human responses to the same prompts.

Because RAID is imbalanced (with fewer human than model-generated examples), we downsample each model's generated subset to 26,700 examples (the minimum across selected models) and use all available human examples (12,900). We split each dataset into 90% training, 5% validation, and 5% test. Throughout our experiments with decoder-only models, we discard prompts, as in practice, it is more likely that a text sample is observed without access to the prompt that generated it.

we use LLAMA 3.1 8B (Grattafiori et al., 2024), COHERE(c4ai-command-r7b-12-2024)(Cohere et al., 2025), GPT-NEO (Black et al., 2021), and MISTRAL(Ministral-3-8B-Instruct-2512) (Jiang et al., 2023) to obtain the attribution as approximate models that have generated the RAID and HC3 datasets.

3.2 Evaluation Metrics

For the evaluation metric, we report Accuracy, Precision, Recall, F1, Area Under Curve, and True/false-positive Rate at a fixed low false-positive operating point, namely $\mathrm{TPR@FPR} = 0.01$. Here $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$ and $\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}}$; thus $\mathrm{FPR} = 0.01$ corresponds to falsely labeling 1% of human-written samples as AI-generated. This is the critical "high-precision" regime. In academic or forensic settings, false positives (accusing a human of using AI) are unacceptable. A detector must have a high TPR at a very low FPR to be deployable (Ayoobi et al., 2025).

3.3 Baseline Models

For the encoder–decoder setting, our baseline is a custom 3-layer Transformer-base classifier trained on paired source–target text. For the decoder-only setting, we report results from Fast-DetectGPT(Curvature) (Bao et al., 2023) and a RoBERTa-based detector released by SuperAnnotate.[3]

3.4 Results

We train and evaluate our detectors under four dataset configurations. In all experiments, we use a class-balanced entropy loss to mitigate the imbalance between human- and AI-generated samples.

Marian-MT. Table 1 reports results for AI-translation detection across three Marian MT-language pairs. Using attribution maps alone (CNN), our method consistently outperforms the text-only baseline for all directions, with gains of +3.6 F1 for ar-en (74.6 vs. 71.0), +6.7 for de–en (81.5 vs. 74.8), and +8.3 for fr-en (85.1 vs. 76.8). Adding target-text features (CNN+text) yields a further, but smaller, improvement over CNN in every case (+2.0, +1.7, and +1.4 F1, respectively), suggesting that most discriminative signal is already captured by the attribution structure, while text provides complementary information.

RAID Dataset. Then we move on to the decoder-only datasets and models. In the second configuration, we train a separate detector for each generator family in RAID, using only samples produced by the corresponding model (and the matched human texts). We call this setting *individual*. In a third configuration, we train on a unified mixture of all RAID generators except Mistral, while reserving Mistral exclusively for testing. To control for training set size, we subsample each included generator's generated text to match the per-model training budget used in the *individual* setting. This split assesses cross-generator generalization, i.e., whether representations learned from a subset of generator families transfer to an unseen model at test time. We call this setting *unified*.

In the *individual* setting (Table 2), AEYEDE achieves near-ceiling performance across all generator families (F1 $\geq$ 96.6; AUC $\geq$ 98.9), substantially outperforming both baselines. The gains are especially pronounced for GPT-Neo and Mistral, where RoBERTa and Curvature remain in the low-to-mid 80s F1, while AEYEDE exceeds 96 F1.

In the *unified* setting (Table 3), the performance drops for all methods, indicating that pooled training introduces a stronger distribution shift. RoBERTa is most stable under unified (F1 $\approx$ 76–78 across models), whereas Curvature degrades markedly (F1 as low as 61.4). AEYEDE remains competitive but shifts toward high precision and lower recall, yielding F1 $\approx$ 74-75 with AUC around 78-80. This precision–recall pattern suggests that attribution structure generalizes conservatively across generator families, and the model makes few false-positive AI calls but misses a larger fraction of AI texts when trained on mixed generators.

Overall, AEYEDE is clearly superior in within-family detection and remains competitive under pooled training, while the curvature-based baseline is consistently the least robust.

[3] https://huggingface.co/SuperAnnotate/ai-detector

Table 1: Performance on Marian-MT generated translation and attribution maps. CNN and CNN+text configurations are based on AEyeED.

	ar–en					de–en					fr–en				
Config	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC
CNN	68.9	63.0	91.4	74.6	77.7	79.0	72.6	93.0	81.5	87.3	83.6	78.0	93.6	85.1	91.0
CNN+text	71.9	65.7	91.9	76.6	81.5	81.1	74.9	93.6	83.2	89.2	85.4	80.3	93.9	86.5	92.7
text (baseline)	61.9	57.3	93.4	71.0	71.5	69.5	63.7	90.5	74.8	77.5	72.4	66.2	91.4	76.8	81.4

Table 2: Performance on RAID (individual).

	AEyeDE					RoBERTa					Curvature				
Generator	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC
Cohere	97.2	97.5	98.4	98.0	99.5	83.3	84.3	92.5	88.2	90.4	83.3	86.3	89.4	87.8	90.9
GPT-Neo	97.2	97.2	98.7	98.0	99.7	75.1	75.0	94.6	83.7	79.6	78.9	98.2	70.0	81.8	79.7
LLaMA	99.2	99.6	99.3	99.4	99.9	96.0	97.9	96.0	97.0	98.3	67.4	67.4	99.9	80.5	78.7
Mistral	95.4	98.1	95.1	96.6	98.9	72.9	72.8	95.8	82.7	79.2	67.7	67.7	100.0	80.7	70.3

Table 3: Performance on RAID (unified).

	AEyeDE					RoBERTa					Curvature				
Generator	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC
Cohere	71.6	94.9	61.2	74.4	79.8	69.7	80.9	72.0	76.2	77.3	57.4	79.0	50.1	61.4	64.0
GPT-Neo	72.3	96.6	61.1	74.8	78.9	69.4	75.8	80.4	78.0	77.1	70.1	93.4	60.0	73.1	77.8
LLaMA	72.0	92.6	63.5	75.4	79.3	70.2	77.5	78.7	78.1	77.9	63.0	82.4	57.5	67.7	69.4
Mistral	71.0	91.5	62.9	74.6	78.4	71.1	79.9	76.6	78.2	78.5	59.3	81.8	51.2	63.0	65.1

Table 4: Performance on HC3.

	AEyeDE					RoBERTa					Curvature				
Model	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC	Acc	Prec	Rec	F1	AUC
GPT-Neo	97.1	98.3	96.4	97.3	99.4	97.7	96.7	98.9	97.8	99.6	98.4	99.8	97.0	98.4	99.2

HC3 Dataset. Finally, we report the results of HC3 with attributions extracted from GPT-Neo used as the proxy model for attribution extraction. Table 4 reports results on HC3 under the *individual* setting. All three approaches perform strongly on this benchmark (F1 $\geq$ 97.3), suggesting that HC3 is comparatively less challenging than RAID. AEyeDE attains 97.3 F1 with high precision (98.3) and AUC 99.4, while RoBERTa slightly improves recall (98.9) and yields 97.8 F1 (AUC 99.6). The curvature baseline achieves the highest overall scores (98.4 F1; 98.4 accuracy), driven by near-perfect precision (99.8) at high recall (97.0). Taken together, HC3 results indicate that a solely attribution-based approach can be as effective as other approaches for this dataset.

4 Discussion: Patch-level motifs in attribution maps

Building on the attribution encoder E_{attr} (Sec. 2), we investigated whether the detector exploits *localized* and *repeatable* visual motifs in attribution maps that are characteristic of human-written versus AI-generated text. Recall that after the final convolutional stage, each block $A_k \in \mathbb{R}^{128\times128}$ is mapped to a feature map of spatial size 16×16 with 256 channels. We denote this last feature map by

$$F_k \in \mathbb{R}^{256\times16\times16}.$$

Because $128/16 = 8$, each feature map cell (u, v) corresponds to an 8×8 patch of the original block A_k(depicted on 1). Let $P_{k,u,v} \in \mathbb{R}^{8\times8}$ be this patch, and define its representation, obtained after the last CNN convolutional stage, as:

$$z_{k,u,v} \;=\; F_k[:, u, v] \in \mathbb{R}^{256}.$$

We denote $\{z_{k,u,v}\}$ as an embedding space of patches produced by the detector model.

Patch selection and clustering. To avoid padding-only and non-informative constant patches, we only keep patches that (i) correspond to non-padding entries under the mask $M = m_x m_y^\top$, and (ii) pass a minimal informativeness threshold based on mean and standard deviation inside the patch. We keep patches with $|\mu_{k,u,v}| > \tau_\mu(0.01)$ and $\sigma_{k,u,v} > \tau_\sigma(0.001)$. Then, we cluster their retained embeddings $\{z_{k,u,v}\}$ using HDBSCAN, producing assignments of each patch to the corresponding cluster

$$c_{k,u,v} \in \{1, \dots, C\} \cup \{-1\}. \tag{15}$$

where -1 is unclustered noise. Each cluster can be interpreted as a *motif family*: a set of patches that the trained encoder maps to nearby representations.

Sample-normalized motif rates. A direct comparison of raw motif counts is complicated by two sources: (i) datasets can be class-imbalanced (e.g., RAID), and (ii) the number of valid retained after preprocessing patches varies greatly across samples due to length and informativity selection. To obtain a metric that is comparable across samples, for each sample s we denote by $\mathcal{P}(s)$ the set of all retained patches extracted from that sample (across all diagonal blocks), and define the per-sample *motif rate* for cluster c as

$$r_s(c) = \frac{1}{|\mathcal{I}_s|} \sum_{(k,u,v)\in\mathcal{I}_s} \mathbb{I}[c_{k,u,v} = c], \tag{16}$$

There $\mathcal{I}_s$ is the set of retained patch indices (k, u, v) from sample s, and let $c_{k,u,v}$ be the patch-cluster assignment. That is, $r_s(c)$ is the fraction of all retained patches in sample s that belong to cluster c.

For a group of samples $\mathcal{S}_g$ (defined below), we summarize prevalence by the mean rate

$$\bar{r}_g(c) = \frac{1}{|\mathcal{S}_g|} \sum\nolimits_{s\in\mathcal{S}_g} r_s(c). \tag{17}$$

In our analysis, we report $\bar{r}_g(c)$ for 2 groups: $\mathcal{S}_{\text{gold human}}$ and $\mathcal{S}_{\text{gold machine}}$. Intuitively, a value such as $\bar{r}_{\text{gold machine}}(c) = 0.01$ means that, on average, *1%* of all retained patches in a machine-labeled sample belong to a cluster c; a difference of 0.15 corresponds to a 15% shift in a an average patch share.

Motif discriminativeness across datasets and proxy models. Given gold-labeled groups $\mathcal{S}_{\text{gold human}}$ and $\mathcal{S}_{\text{gold machine}}$, we quantify how strongly a motif cluster separates classes via the difference in mean motif rates

$$\Delta r(c) = \bar{r}_{\text{gold machine}}(c) - \bar{r}_{\text{gold human}}(c). \tag{18}$$

Using the aggregate statistics (top-3 clusters by $|\Delta r(c)|$ per setting), we find that the most discriminative motif often shifts by several percentage points in average patch share, and in some cases by more than 10 points. Table 5 reports the top-1 cluster per dataset/proxy/preprocessing variant, including bootstrap confidence intervals for $\Delta r(c)$ and a permutation-test p-value. In HC3, the strongest motif (cid=24) is *machine-enriched*, increasing from 9.7% (gold human) to 20.7% (gold machine), $\Delta r = +11.0$ points with a tight 95% CI $[9.3, 12.7]$. In contrast, for RAID the top motifs are consistently *human-enriched* under all tested proxy models and both preprocessing variants (e.g., $\Delta r = -10.5$ points for COHERE under the unified variant). Qualitatively, the corresponding example patches (Fig. 2i) show similar, repeatable local structures: for human-leaning patch clusters, "islands" and isolated flashes of attention are observed across all datasets and models, while the only identified heavy machine-inclined cluster of HC3/GPT-NEO exhibits a prevalence of the horizontal bands.The notable outlier is GPT-Neo for RAID, which is explained by HDBSCAN producing a very low amount of dense clusters(n = 4) resulting in a high intra-class variance. Such observations support the interpretation of clusters as visually coherent "motif families".

Patch-wise saliency and ablation are computed per retained patch. To relate motif *prevalence* to motif *importance*, we assign each retained patch an (i) Grad-CAM(Selvaraju et al., 2019) score and (ii) a zeroing-ablation score, and then aggregate these scores by cluster. Concretely, for each retained patch index $(k, u, v) \in \mathcal{I}_s$ we compute: (i) a Grad-CAM activation $g_{s,k,u,v}$ at the last convolutional stage (the same 16×16 grid that defines the patches), and (ii) an ablation-induced logit change $\delta_{s,k,u,v}$ obtained by zeroing the corresponding 8×8 region in the input block A_k and re-evaluating the detector. We then summarize cluster-level importance by averaging over all occurrences assigned

Table 5: Top-1 motif cluster by absolute class-rate gap $|\Delta r(c)|$ for each dataset/setting. Rates are mean per-sample motif shares (%), and Δr is reported in percentage points (machine minus human). Confidence intervals are 95% bootstrap CIs over samples; p is from a label-permutation test.

Dataset	Setting	Proxy-model	cid	Human $\bar{r}_g$	Machine $\bar{r}_g$	Δr	95% CI	p
HC3	individual	GPT-NEO 2.7B	24	9.7	20.7	+11.0	[9.3, 12.7]	<0.001
RAID	unified	COHERE Cmd-R 7B	13	80.0	69.5	-10.5	[-12.9, -8.2]	<0.001
RAID	individual	MISTRAL 7B	23	63.5	54.8	-8.7	[-11.1, -6.2]	<0.001
RAID	individual	GPT-NEO 2.7B	4	87.6	81.1	-6.5	[-9.0, -4.1]	<0.001
RAID	individual	LLAMA 3.1 8B	12	7.5	3.1	-4.4	[-4.8, -3.9]	<0.001
RAID	unified	GPT-NEO 2.7B	37	11.6	8.5	-3.0	[-4.1, -2.0]	<0.001
RAID	unified	MISTRAL 7B	39	7.3	5.1	-2.2	[-2.9, -1.6]	<0.001
RAID	individual	COHERE Cmd-R 7B	8	82.1	79.9	-2.1	[-3.7, -0.6]	0.031
RAID	unified	LLAMA 3.1 8B	16	9.5	7.6	-2.0	[-2.4, -1.5]	<0.001

to cluster c:

$$\bar{g}(c) = \mathbb{E}[g_{s,k,u,v} \mid c_{k,u,v} = c]\,, \qquad \bar{\delta}(c) = \mathbb{E}[\delta_{s,k,u,v} \mid c_{k,u,v} = c]\,. \tag{19}$$

Here $\bar{\delta}(c)$ captures the *average marginal effect* of removing a single motif instance, while $\bar{r}_g(c)$ captures *how frequently* the motif occurs in a given group.

Why prevalence need not correlate with saliency or ablation. Empirically, we observe that clusters with the largest $|\Delta r(c)|$ are not necessarily the clusters with the highest $\bar{g}(c)$ or $|\bar{\delta}(c)|$. Aggregated correlations across all datasets and models are weak and unstable: Across top-15 clusters for GRAD-CAM Pearson $\rho = 0.143 \pm 0.23$, Spearman $\rho = 0.10 \pm 0.29$; for Zero-ablation Pearson $\rho =$ -0.03 $\pm$ 0.36, Spearman $\rho =$ -0.09 $\pm$ 0.25. Across top-3 clusters correlation becomes moderate for GRAD-CAM but still unstable: Pearson $\rho = 0.373 \pm 0.66$, Spearman $\rho = 0.37 \pm 0.54$; while slightly improves for Zero-ablation, Pearson $\rho = 0.09 \pm 0.67$, Spearman $\rho = 0.12 \pm 0.64$) We attribute this mismatch to following reasons: First, a dataset imbalance can decouple prevalence from importance. Taking into an account the significant imbalance of used datasets - RAID subset and HC3, training under skewed priors biases the detector toward features that optimize empirical risk for the majority class, so a motif may be strongly pronounced in $\Delta r(c)$ yet have muted average Grad-CAM/ablation effects. Second, for Grad-CAM and zeroing-ablation the same motif family may be decisive only in certain positions or alongside other motifs; averaging $\bar{g}(c)$ and $\bar{\delta}(c)$ across all occurrences can weaken these conditional effects.

Taken together, these findings suggest that motif analysis is best interpreted as a complementary view: $\Delta r(c)$ reveals dataset and proxy-model dependent differences in local attention map structure, while $\bar{g}(c)$ and $\bar{\delta}(c)$ reflect how the trained detector *uses* (or ignores) individual motif instances at decision time conditioned on the training data.

Implications Overall, the presence of statistically reliable shifts in patch motif rates between gold machine and gold human groups (Table 5) supports our hypothesis that the internal dynamics of a proxy G_θ induce detectable structure in attention-based attribution maps beyond surface-level text statistics. At the same time, the weak alignment between prevalence and saliency cautions against interpreting frequent motifs as predictions of the model behaviour, instead, they appear to function as stable, repeatable signatures that the detector can exploit in combination with other cues.

5 Conclusion

We presented AEYEDE, an attribution-based framework for AI-generated text detection that leverages attention-derived attribution maps from a proxy Transformer as structured evidence. Across both encoder–decoder translation benchmarks (WMT14, UN) and decoder-only generation datasets (HC3, RAID), AEYEDE achieves competitive performance and shows strong within-family detection, while remaining robust under generator shift in the unified RAID setting. Beyond accuracy, we provide an analysis of localized attribution patterns and show that they systematically differ between human and AI generated text. Overall, our results suggest that internal attribution behavior offers a complementary and effective signal for reliable authorship detection, motivating further work on broader robustness settings and alternative attribution sources.

Limitations

Our study has some limitations. First, the proposed framework assumes *white-box* access to a proxy Transformer model in order to extract attention-based attribution maps. While this assumption may limit applicability in fully black-box settings, our experiments indicate that attribution patterns generalize across generator families, suggesting that exact access to the true generator is not strictly required.

Second, for decoder-only models, we summarize attribution structure by extracting fixed-size blocks along the main diagonal, motivated by prior evidence that informative self-attention dynamics are concentrated locally. This design prioritizes computational efficiency and interpretability; exploring richer off-diagonal structures remains an interesting direction for future work.

Furthermore, our implementation focuses on attention-based attributions, which offer a favorable trade-off between informativeness and computational cost for large models and long sequences. Investigating alternative attribution methods, such as gradient-based saliency, may further enrich the analysis but is left for future work due to their higher computational overhead.

Finally, while our evaluation spans multiple datasets, architectures, and LLM families, it does not exhaustively cover all possible transformation or editing scenarios. Extending the evaluation to additional settings, such as human–AI co-authoring or adversarial rewriting, remains a promising direction for future research.

Use of AI Assistance

In preparing this work, we used AI-assisted tools for code completion with GitHub Copilot and language editing, such as spell checking and stylistic revisions with Grammarly and ChatGPT. The authors take full responsibility for the scientific content, analyses, and conclusions presented in this work.

References

Shadi Abudalfa, Saad Ezzini, Ahmed Abdelali, Hamza Alami, Abdessamad Benlahbib, Salmane Chafik, Mo El-Haj, Abdelkader El Mahdaouy, Mustafa Jarrar, Salima Lamsiyah, and 1 others. 2025. The arageneval shared task on arabic authorship style transfer and ai generated text detection. In *Proceedings of The Third Arabic Natural Language Processing Conference: Shared Tasks*, pages 1–13.

Hyeseon Ahn, Shinwoo Park, Suyeon Woo, and Yo-Sub Han. 2025. Ditto: A spoofing attack framework on watermarked llms via knowledge distillation. *arXiv preprint arXiv:2510.10987*.

Firoj Alam, Preslav Nakov, Nizar Habash, Iryna Gurevych, Shammur Chowdhury, Artem Shelmanov, Yuxia Wang, Ekaterina Artemova, Mucahid Kutlu, and George Mikros, editors. 2025. *Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect)*. International Conference on Computational Linguistics, Abu Dhabi, UAE.

Muhammad Zain Ali, Yuxia Wang, Bernhard Pfahringer, and Tony C Smith. 2025. Detection of human and machine-authored fake news in Urdu. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3419–3428, Vienna, Austria. Association for Computational Linguistics.

Navid Ayoobi, Sadat Shahriar, and Arjun Mukherjee. 2025. Beyond easy wins: A text hardness-aware benchmark for llm-generated text detection. *arXiv preprint arXiv:2507.15286*.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2023. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *arXiv preprint arXiv:2310.05130*.

Kyle Bittle and Omar El-Gayar. 2025. Generative ai and academic integrity in higher education: A systematic review and research agenda. *Information*, 16(4):296.

Sid Black, Gao Leo, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, and 1 others. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pages 12–58.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, and 1 others. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Team Cohere, Aakanksha, Arash Ahmadian, Marwan Ahmed, Jay Alammar, Yazeed Alnumay, Sophia Althammer, Arkady Arkhangorodsky, Viraat Aryabumi, Dennis Aumiller, Raphaël Avalos, Zahara Aviv, Sammie Bae, Saurabh Baji, Alexandre Barbet, Max Bartolo, Björn Bebensee, Neeral Beladia, Walter Beller-Morales, and 207 others. 2025. Command a: An enterprise-ready large language model. *Preprint*, arXiv:2504.00698.

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.

Xinyue Cui, Johnny Wei, Swabha Swayamdipta, and Robin Jia. 2025. Robust data watermarking in language models by injecting fictitious knowledge. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14292–14306.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Liam Dugan, Alyssa Hwang, Filip Trhlík, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. RAID: A shared benchmark for robust evaluation of machine-generated text detectors. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12463–12492, Bangkok, Thailand. 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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12763–12771.

Markus Frohmann, Gabriel Meseguer-Brocal, Markus Schedl, and Elena V. Epure. 2025. Double entendre: Robust audio-based AI-generated lyrics detection via multi-view fusion. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1914–1926, Vienna, Austria. Association for Computational Linguistics.

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

Wooyoung Go, Hyoungshick Kim, Alice Oh, and Yongdae Kim. 2025a. XDAC: XAI-driven detection and attribution of LLM-generated news comments in Korean. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22728–22750, Vienna, Austria. Association for Computational Linguistics.

Wooyoung Go, Hyoungshick Kim, Alice Oh, and Yongdae Kim. 2025b. Xdac: Xai-driven detection and attribution of llm-generated news comments in korean. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22728–22750.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2024. Mgtbench: Benchmarking machine-generated text detection. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 2251–2265.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: Robust ai-text detection via adversarial learning. *Advances in neural information processing systems*, 36:15077–15095.

Beining Huang, Du Su, Fei Sun, Qi Cao, Huawei Shen, and Xueqi Cheng. 2025a. Low-entropy watermark detection via bayes' rule derived detector. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14330–14344.

He Huang, Nan Sun, Massimiliano Tani, Yu Zhang, Jiaojiao Jiang, and Sanjay Jha. 2025b. Can llm-generated misinformation be detected: A study on cyber threat intelligence. *Future Generation Computer Systems*, 173:107877.

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

Michael Ivanitskiy, Cecilia Diniz Behn, and Samy Wu Fung. Motifs in attention patterns of large language models. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Kaijie Jiao, Quan Wang, Licheng Zhang, Zikang Guo, and Zhendong Mao. 2025. M-RangeDetector: Enhancing generalization in machine-generated text detection through multi-range attention masks. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8971–8983, Vienna, Austria. Association for Computational Linguistics.

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

Kalpesh Krishna, Yapei Chang, John Wieting, and Mohit Iyyer. 2022. RankGen: Improving text generation with large ranking models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 199–232, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kristian Kuznetsov, Laida Kushnareva, Anton Razzhigaev, Polina Druzhinina, Anastasia Voznyuk, Irina Piontkovskaya, Evgeny Burnaev, and Serguei Barannikov. 2025. Feature-level insights into artificial text detection with sparse autoencoders. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25727–25748, Vienna, Austria. Association for Computational Linguistics.

Salima Lamsiyah, Saad Ezzini, Abdelkader El Mahdaouy, Hamza Alami, Abdessamad Benlahbib, Samir El Amrany, Salmane Chafik, and Hicham Hammouchi. 2025. M-daigt: A shared task on multi-domain detection of ai-generated text. *arXiv preprint arXiv:2511.11340*.

Jiatao Li and Xiaojun Wan. 2025. Who writes what: Unveiling the impact of author roles on AI-generated text detection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26620–26658, Vienna, Austria. Association for Computational Linguistics.

Yuanfan Li, Zhaohan Zhang, Chengzhengxu Li, Chao Shen, and Xiaoming Liu. 2025. Iron sharpens iron: Defending against attacks in machine-generated text detection with adversarial training. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3091–3113, Vienna, Austria. Association for Computational Linguistics.

Aiwei Liu, Leyi Pan, Yijian Lu, Jingjing Li, Xuming Hu, Xi Zhang, Lijie Wen, Irwin King, Hui Xiong, and Philip Yu. 2024a. A survey of text watermarking in the era of large language models. *ACM Computing Surveys*, 57(2):1–36.

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*.

Zeyan Liu, Zijun Yao, Fengjun Li, and Bo Luo. 2024b. On the detectability of chatgpt content: Benchmarking, methodology, and evaluation through the lens of academic writing. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 2236–2250.

Xinyang Lu, Jingtan Wang, Zitong Zhao, Zhongxiang Dai, Chuan-Sheng Foo, See Kiong Ng, and Bryan Kian Hsiang Low. 2025. Wasa: Watermark-based source attribution for large language model-generated data. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 23791–23824.

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, Bangkok, Thailand. Association for Computational Linguistics.

Dominik Macko, Jakub Kopál, Robert Moro, and Ivan Srba. 2025. MultiSocial: Multilingual benchmark of machine-generated text detection of social-media texts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 727–752, Vienna, Austria. Association for Computational Linguistics.

Minjia Mao, Dongjun Wei, Zeyu Chen, Xiao Fang, and Michael Chau. 2025. Watermarking large language models: An unbiased and low-risk method. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7939–7960.

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.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*.

Georg Niess and Roman Kern. 2025. Ensemble watermarks for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2903–2916.

OpenAI. 2023. Chatgpt. https://chat.openai.com. Large language model.

Andrea Pedrotti, Michele Papucci, Cristiano Ciaccio, Alessio Miaschi, Giovanni Puccetti, Felice Dell'Orletta, and Andrea Esuli. 2025. Stress-testing machine generated text detection: Shifting language models writing style to fool detectors. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3010–3031, Vienna, Austria. Association for Computational Linguistics.

Xinlin Peng, Ying Zhou, Ben He, Le Sun, and Yingfei Sun. 2023. Hidding the ghostwriters: An adversarial evaluation of ai-generated student essay detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10406–10419.

Jiawen Qi, Chang Gao, Zhaochun Ren, and Qinyu Chen. 2025. Deltallm: A training-free framework exploiting temporal sparsity for efficient edge llm inference. *arXiv preprint arXiv:2507.19608*.

Rafael Alberto Rivera Soto, Barry Y. Chen, and Nicholas Andrews. 2025. Mitigating paraphrase attacks on machine-text detection via paraphrase inversion. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4421–4433, Vienna, Austria. Association for Computational Linguistics.

Gabriele Sarti, Nils Feldhus, Ludwig Sickert, and Oskar van der Wal. 2023. Inseq: An interpretability toolkit for sequence generation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 421–435, Toronto, Canada. Association for Computational Linguistics.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2019. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359.

Lujia Shen, Xuhong Zhang, Shouling Ji, Yuwen Pu, Chunpeng Ge, Xing Yang, and Yanghe Feng. 2023. Textdefense: Adversarial text detection based on word importance entropy. *arXiv preprint arXiv:2302.05892*.

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.

Vasiliki Tassopoulou, George Retsinas, and Petros Maragos. 2021. Enhancing handwritten text recognition with n-gram sequence decomposition and multitask learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10555–10560. IEEE.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT – building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Irina Tolstykh, Aleksandra Tsybina, Sergey Yakubson, and Maksim Kuprashevich. 2025. Llmtrace: A corpus for classification and fine-grained localization of ai-written text. *arXiv preprint arXiv:2509.21269*.

Adaku Uchendu, Zeyu Ma, Thai Le, Rui Zhang, and Dongwon Lee. 2021. Turingbench: A benchmark environment for turing test in the age of neural text generation. *arXiv preprint arXiv:2109.13296*.

Ashok Urlana, Aditya Saibewar, Bala Mallikarjunarao Garlapati, Charaka Vinayak Kumar, Ajeet Singh, and Srinivasa Rao Chalamala. 2024. TrustAI at SemEval-2024 task 8: A comprehensive analysis of multi-domain machine generated text detection techniques. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 927–934, Mexico City, Mexico. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, and Thomas Arnold. 2024a. SemEval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 2057–2079, Mexico City, Mexico. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024b. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1369–1407, St. Julian's, Malta. Association for Computational Linguistics.

Zongqi Wang, Tianle Gu, Baoyuan Wu, and Yujiu Yang. 2025. MorphMark: Flexible adaptive watermarking for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4842–4860, Vienna, Austria. Association for Computational Linguistics.

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

Junchao Wu, Runzhe Zhan, Derek Wong, Shu Yang, Xinyi Yang, Yulin Yuan, and Lidia Chao. 2024. Detectrl: Benchmarking llm-generated text detection in real-world scenarios. *Advances in Neural Information Processing Systems*, 37:100369–100401.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2023. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*.

Xiaowei Zhu, Yubing Ren, Yanan Cao, Xixun Lin, Fang Fang, and Yangxi Li. 2025. Reliably bounding false positives: A zero-shot machine-generated text detection framework via multiscaled conformal prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12298–12319, Vienna, Austria. Association for Computational Linguistics.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

A Related Work

AI-Generated Text Detection. Research on detecting machine-generated text has accelerated alongside the rapid progress and deployment of LLMs (Wu et al., 2025). Existing approaches can be categorized into (i) *surface-statistical* and (ii) *likelihood-based* detectors, (iii) *supervised neural classifiers*, (iv) *watermarking and source attribution*, and (v) *LLM-based* meta-detectors. Surface-statistical methods exploit distributional artifacts such as perplexity, burstiness, or n-gram irregularities, often providing lightweight but increasingly fragile signals as generators improve (Gehrmann et al., 2019; Ippolito et al., 2020; Shen et al., 2023; Tassopoulou et al., 2021; Krishna et al., 2022). Complementarily, likelihood-based methods probe the generator's probability landscape: DetectGPT identifies machine text by measuring curvature via perturbations (Mitchell et al., 2023), and related work improves efficiency and robustness through faster perturbation schemes (Bao et al., 2023). These lines of work capture model-specific statistical footprints, but can degrade as LLMs are optimized to match human-like distributions.

Neural Detectors, Robustness, and Generalization. Supervised detectors typically fine-tune Transformer encoders (e.g., BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019)) on labeled human vs. machine text, achieving strong in-domain performance but often suffering under domain shift and unseen generators (Uchendu et al., 2021; Wang et al., 2024b). Robustness has become a central focus: training on diverse decoding strategies improves resilience (Ippolito et al., 2020), adversarial training frameworks such as IRON harden detectors against evasion (Li et al., 2025), and Radar explicitly targets robustness via adversarial learning (Hu et al., 2023). Recent methods further aim to improve out-of-distribution behavior and reliability guarantees, e.g., by shaping attention over multiple receptive ranges (Jiao et al., 2025) or bounding false positives with conformal prediction in zero-shot settings (Zhu et al., 2025). Interpretability for detectors is also receiving attention: feature-level analyses using sparse autoencoders help reveal which latent patterns separate machine and human text (Kuznetsov et al., 2025), while downstream applications increasingly require multilingual and domain-specific robustness (Ali et al., 2025) and fine-grained settings such as human–AI co-authorship (Su et al., 2025).

Watermarking and Source Attribution. Watermarking aims to embed detectable signals into generated text, enabling attribution when generation-side cooperation is available (Liu et al., 2024a). Early and widely adopted schemes include token-list or "soft" watermarks that bias sampling (Kirchenbauer et al., 2023), while subsequent work explored alternative embedding mechanisms and detection rules, including entropy- or Bayesian-inspired detectors (Lu et al., 2024; Huang et al., 2025a) and more adaptive watermark designs such as MorphMark (Wang et al., 2025). Recent studies further examine watermark ensembles (Niess and Kern, 2025), watermark-based source attribution (e.g., WASA) (Lu et al., 2025), and approaches that reduce bias and risk (Mao et al., 2025). However, watermarking remains challenged by post-editing and paraphrasing (Liu et al., 2024a), motivating defenses such as paraphrase inversion (Rivera Soto

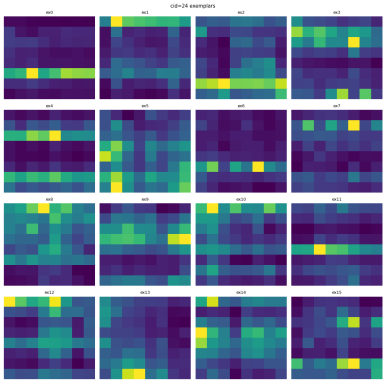
et al., 2025) and robustness through injected "fictitious knowledge" signals (Cui et al., 2025). Adversarial settings also reveal vulnerabilities: DITTO formalizes spoofing attacks against watermarked LLMs via knowledge distillation (Ahn et al., 2025), underscoring the need for evaluation under realistic transformation and attack pipelines.

LLMs as Detectors and Explainable Attribution. Beyond classical detectors, LLMs are increasingly used as meta-detectors and critics of generated content, reflecting a trend toward black-box and instruction-following detection pipelines (Wang et al., 2024b). Recent work expands from binary detection to attribution and explanation, e.g., XDAC provides XAI-driven detection and attribution for Korean news comments (Go et al., 2025a,b), and studies of detectability highlight how author intent and role can affect detection outcomes (Li and Wan, 2025). Together, these directions emphasize that practical detection increasingly requires robustness, reliability, and interpretable evidence—not only raw accuracy.

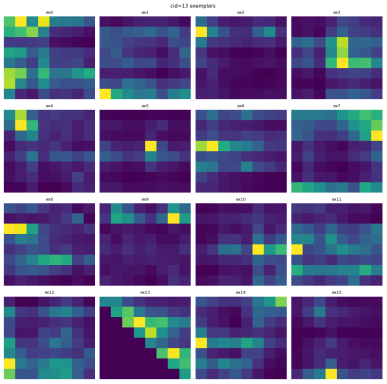
Benchmarks and Shared Tasks. Progress in AI-text detection is tightly coupled with benchmarks that stress generalization across domains, languages, and attack conditions. Widely used datasets include HC3 (Guo et al., 2023), MGT-Bench (He et al., 2024), WritingPrompts (Bao et al., 2023), RAID (Dugan et al., 2024), and adversarial extensions such as Adv-HC3 (Peng et al., 2023); additional resources target broader settings such as BUST (Cornelius et al., 2024) and LLMTRACE (Tolstykh et al., 2025). Beyond text-only benchmarks, MultiSocial supports multilingual social-media detection (Macko et al., 2025), Double Entendre introduces a multimodal audio-lyrics setting (Frohmann et al., 2025), and stress-test benchmarks systematically perturb style to probe brittleness (Pedrotti et al., 2025). Shared tasks further standardize evaluation and accelerate methodology: SemEval-2024 Task 8 targets black-box, multilingual, and multidomain detection (Wang et al., 2024a), with system analyses such as TrustAI highlighting practical modeling choices (Urlana et al., 2024). Community efforts such as the GenAIDetect workshop at COLING 2025 (Alam et al., 2025) and domain-focused shared tasks and datasets (e.g., M-DAIGT for news and academic writing (Lamsiyah et al., 2025), and AraGenEval for Arabic settings (Abudalfa et al., 2025)) reflect increasing emphasis on robustness, multilinguality, and real-world constraints. These benchmarks and tasks collectively motivate detectors that generalize across generator families while offering transparent, verifiable evidence for their decisions.

Positioning of Our Work. In contrast to prior detection approaches that primarily rely on surface statistics, likelihood perturbations, or end-to-end text representations, our method explicitly targets the *internal attribution behavior* of Transformer models as a detection signal. Unlike watermarking, it does not require model-side cooperation and remains applicable to legacy and closed-source generators; unlike supervised neural detectors, it does not depend solely on lexical or stylistic cues that are vulnerable to paraphrasing and distribution shift. While recent work has begun to explore interpretability and attribution for analysis or explanation, our framework is, to our knowledge, the first to operationalize attention-based attribution maps as structured inputs to a dedicated detection architecture. By combining multi-scale convolutional modeling of attribution patterns with attention pooling and faithfulness-driven explanations, our approach directly leverages model-internal processing dynamics, offering improved robustness, cross-model generalization, and interpretable evidence for detection decisions.

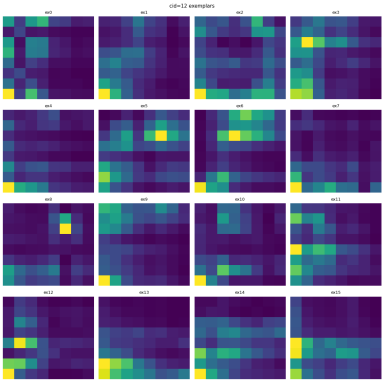
B TPR-FPR



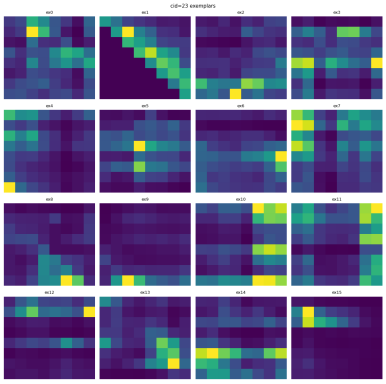
(a) HC3 (individual), proxy: GPT-Neo. Top cluster by $\Delta\bar{r}$ (machine-skewed).



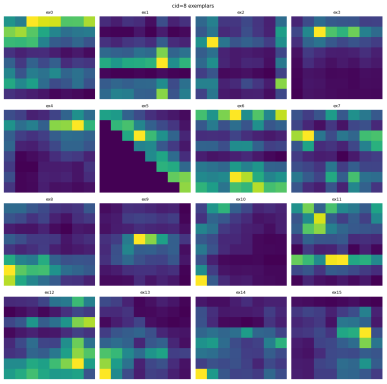
(b) RAID (individual), proxy: Cohere. Top cluster by $\Delta\bar{r}$ (human-skewed).



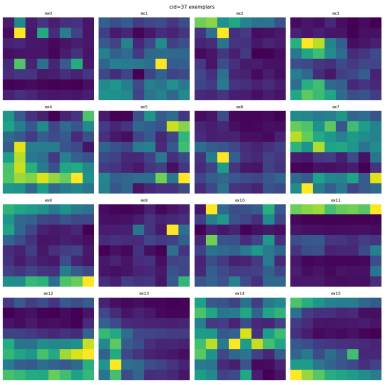
(c) RAID (individual), proxy: LLaMA. Top cluster by $\Delta\bar{r}$ (human-skewed).



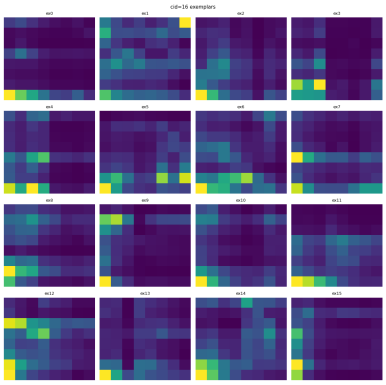
(d) RAID (undividual), proxy: Mistral. Top cluster by $\Delta\bar{r}$ (human-skewed).



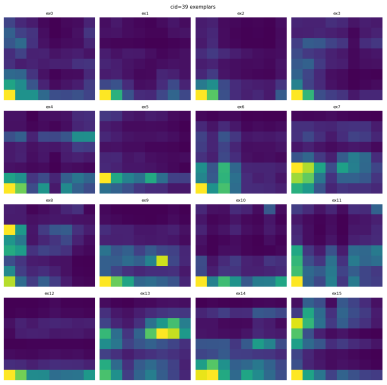
(e) RAID (individual), proxy: Cohere. Top cluster by $\Delta\bar{r}$ (human-skewed).



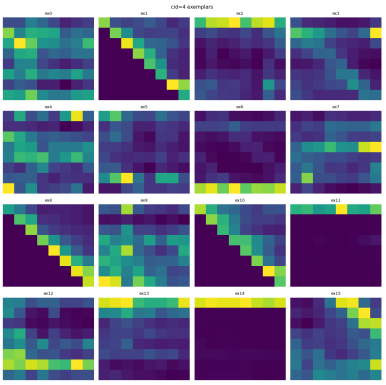
(f) RAID (unified), proxy: GPT-Neo. Top cluster by $\Delta\bar{r}$ (human-skewed).



(g) RAID (unified), proxy: LLaMA. Top cluster by $\Delta\bar{r}$ (human-skewed).

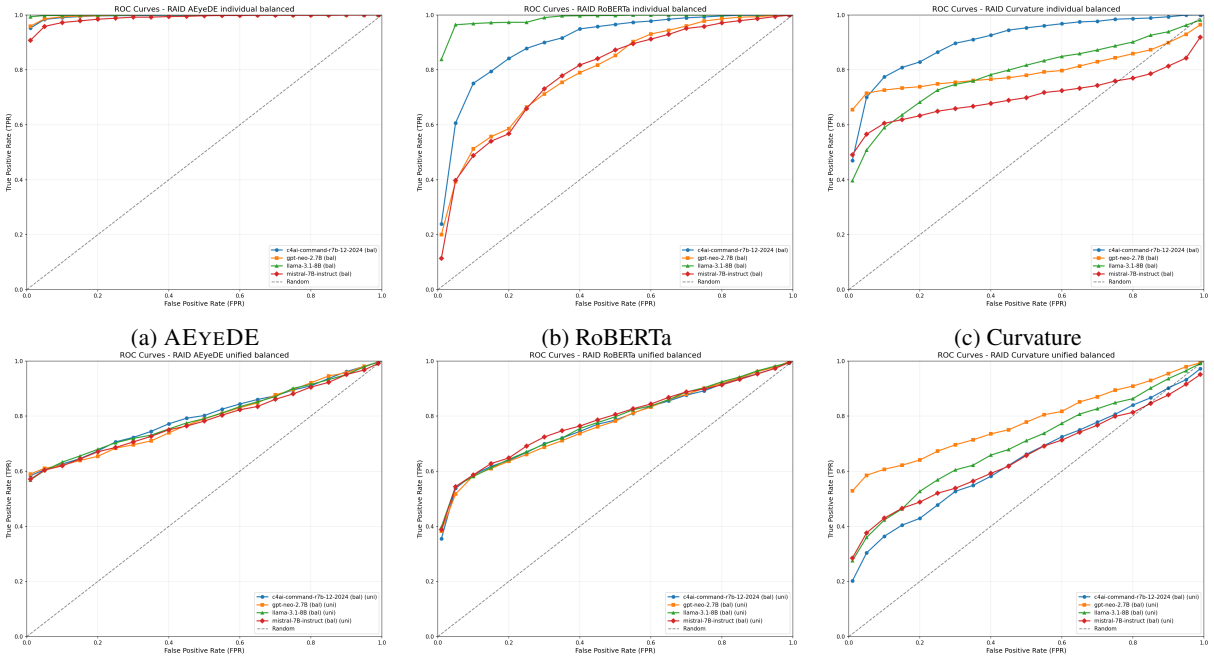


(h) RAID (unified), proxy: Mistral. Top cluster by $\Delta\bar{r}$ (human-skewed).



(i) RAID (individual), proxy: GPT-Neo. Top cluster by $\Delta\bar{r}$ (human-skewed)

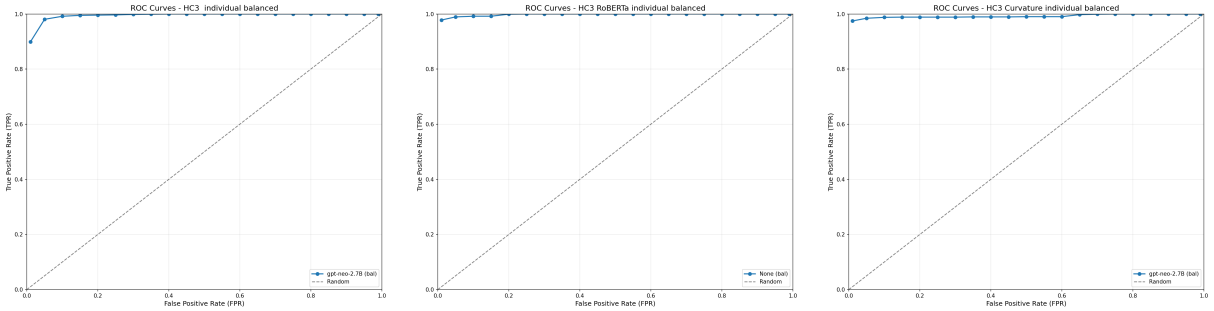
Figure 2: Examples of the top motif cluster (by absolute mean prevalence gap $\Delta\bar{r}$ between gold-machine and gold-human) for each dataset/proxy-model configuration. Each panel shows representative 8×8 patches (z-normalized) from the corresponding cluster.



(a) AEyeDE (b) RoBERTa (c) Curvature

(d) AEyeDE (e) RoBERTa (f) Curvature

Figure 3: ROC curves on RAID (balanced). Top row: *individual*. Bottom row: *unified*. Columns follow the table order: AEyeDE, RoBERTa, Curvature.



(a) AEyeDE (b) RoBERTa (c) Curvature

Figure 4: ROC curves on HC3. Columns follow the table order: AEyeDE, RoBERTa, Curvature.