

LLMTrace: A Corpus for Classification and Fine-Grained Localization of AI-Written Text

Irina Tolstykh, Aleksandra Tsybina, Sergey Yakubson, Maksim Kuprashevich

SALUTEDEV LLC, Tashkent, Uzbekistan

Correspondence: irinakr4snova@gmail.com

Abstract

The widespread use of human-like text from Large Language Models (LLMs) necessitates the development of robust detection systems. However, progress is limited by a critical lack of suitable training data; existing datasets are often generated with outdated models, are predominantly in English, and fail to address the increasingly common scenario of mixed human-AI authorship. Crucially, while some datasets address mixed authorship, none provide the character-level annotations required for the precise localization of AI-generated segments within a text. To address these gaps, we introduce LLMTrace, a new large-scale, bilingual (English and Russian) corpus for AI-generated text detection. Constructed using a diverse range of modern proprietary and open-source LLMs, our dataset is designed to support two key tasks: traditional full-text binary classification (human vs. AI) and AI-generated interval detection, facilitated by character-level annotations. We believe LLMTrace will serve as a vital resource for training and evaluating the next generation of more nuanced and practical AI detection models.

1 Introduction

The rapid advancement and widespread adoption of Large Language Models (LLMs) have enabled the generation of human-like text at an unprecedented scale. This text is often so convincing that human evaluators struggle to distinguish it from human writing, with performance frequently hovering around chance levels (Jakesch et al., 2023; Milíčka et al., 2025). This capability, while offering enormous benefits, presents a significant dual-use challenge: the same models that assist in creative writing can also generate misinformation, compromise academic integrity, and automate malicious communication. The unreliability of human detection underscores the critical need for robust, automated systems to maintain a trustworthy digital

ecosystem.

However, the effectiveness of any detection model is fundamentally limited by its training data. A review of the current landscape (Section 2) reveals several critical gaps: existing datasets often rely on outdated LLMs, are predominantly English-centric with a scarcity of resources for languages like Russian, and focus on simple classification, overlooking the increasingly common scenario of mixed human-AI authorship.

Perhaps the most significant gap is the absence of large-scale resources for the task of *localizing* AI-generated segments within these mixed texts. While a few datasets support boundary detection between authorship (Table 1), they are often limited to single transition points or lack the precise, character-level annotations required to train granular, high-resolution detectors.

To address these limitations, we introduce LLMTrace, a new large-scale corpus designed to push the boundaries of AI text detection. We instantiate LLMTrace for English (the most studied target) and Russian (a high-impact under-served language with its own generator families, e.g. GigaChat, YaGPT). The pipeline itself is language-agnostic and extends to new languages given a human corpus and adapted prompts, positioning LLMTrace as an extensible framework rather than a fixed snapshot.

Our work makes two primary contributions. First, we provide a comprehensive classification dataset built from a diverse suite of 38 LLMs across nine domains, utilizing a wide variety of prompt types to create sophisticated and realistic examples. Second, and most crucially, we introduce the large-scale detection dataset, constructed using 31 generators, with precise, character-level annotations for mixed-authorship texts. These mixed examples are created through a robust pipeline involving automated gap-filling in human texts, AI-based continuation of human texts, and meticulous

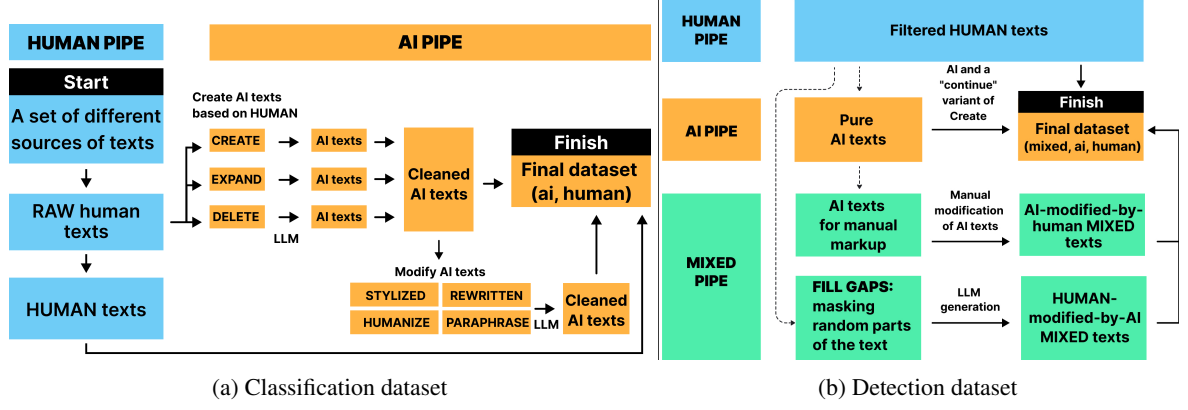


Figure 1: Data curation pipelines: (a) classification builds parallel human/AI corpora; (b) detection aggregates filtered human, pure-AI, and mixed-authorship samples. Dashed arrows mark source texts reused for generation or modification (e.g., “Fill Gaps” inserts AI spans into human texts; manual markup edits AI texts).

manual editing of AI texts by humans, ensuring a high degree of complexity. By making this resource publicly available, we aim to facilitate a new wave of research into more nuanced, practical, and robust AI detection models. The dataset, models, and training code will be publicly available at <https://sweetdream779.github.io/LLMTrace-info/>.

2 Related Works

Robust evaluation of AI text detectors requires diverse benchmarks for two key tasks: binary classification (human vs. AI) and the localization of AI-generated content within mixed-authorship texts. Existing detection resources are summarized in Table 1; a parallel comparison against 17 classification benchmarks is provided in Appendix Table 9.

2.1 Classification Benchmarks

Foundational benchmarks like MAGE (Li et al., 2023), MGTBench (He et al., 2024), BUST (Cornelius et al., 2024), RAID (Dugan et al., 2024), DetectRL (Wu et al., 2024), and HC3 (Guo et al., 2023) have laid important groundwork but are often limited by narrow prompt diversity, a small suite of generator models, or restricted domain coverage. Multilingual resources such as MULTITuDE (Macko et al., 2023), M4GT-Bench (Wang et al., 2024b), and MultiSocial (Macko et al., 2024) have expanded language coverage but are frequently confined to specific domains like news or social media.

The RuATD dataset (Shamardina et al., 2022), currently the only large-scale benchmark for AI text detection in Russian covering 14 generators, spans diverse domains but remains limited to a

single language and lacks full domain coverage.

Domain-specific corpora, while valuable, target narrow areas like Wikipedia edits (WETBench (Quaremba et al., 2025)), scholarly reviews (Peer Review Detection (Yu et al., 2025)), or student essays (GEDE (Gehring and Paaßen, 2025)). Similarly, robustness benchmarks like ESPERANTO (Ayoobi et al., 2024) and SHIELD (Ayoobi et al., 2025) explore adversarial attacks or “humanification” but remain limited in scope. Finally, while datasets like Beemo (Artemova et al., 2024) and MixSet (Zhang et al., 2024) consider mixed authorship, they do so only in a classification setting and at a smaller scale.

We also note the existence of active defense methods such as watermarking (Kirchenbauer et al., 2023); however, our work focuses on passive detection in non-watermarked settings, which remains the prevalent scenario for open-ended text generation.

2.2 Detection Benchmarks

Existing mixed-authorship benchmarks often face structural or domain limitations. M4GT-Bench (Wang et al., 2024b) and RoFT (Dugan et al., 2023) (along with RoFT-chatgpt (Kushnareva et al., 2023)) model only single-transition continuations, failing to capture complex arbitrary boundaries between human and AI segments. While TriBERT (Zeng et al., 2024) supports up to three such transitions, it is confined to student essays. Similarly, CoAuthor (Lee et al., 2022) captures collaborative writing, but its frequent minor edits introduce noise that complicates systematic evaluation.

Table 1: Statistics of the detection datasets. Our LLMTrace dataset includes human, AI, and mixed texts, with the size of the mixed subset noted in parentheses.

Dataset	Dataset Size	# Domains	# Lang	# Gen	# Intervals	Human collab
M4GT-Bench	31.9k	2	1	5	1	✗
RoFT	27.6k	4	1	5	1	✗
RoFT-chatgpt	6.9k	4	1	1	1	✗
TriBERT	17.1k	1	1	1	1-3	✗
CoAuthor	1.4k	2	1	1	arbitrary	✓
LLMTrace (ours)	79,342 (27.7k mixed)	9	2	31	arbitrary	✓

3 LLMTrace: Design Principles and Curation Pipeline

We detail the methodology for creating LLMTrace, a large-scale dataset structured to support binary classification and segment-level detection, designed to ensure diversity, robustness, and real-world applicability. We instantiate this methodology for English and Russian (motivation in Section 1), but every stage of the pipeline (human corpus aggregation, prompt-driven AI generation, mixed-text construction, and post-processing) is language-agnostic and can be applied to additional languages given a new human corpus, adapted prompt templates, and language-tuned filters. The overall pipelines for classification and detection datasets generation are shown in Figure 1.

3.1 Target Tasks and Annotation Schema

Our dataset supports two tasks (examples in Appendix Figure 15). First, **Binary Text Classification** is defined as $\mathcal{D}_{\text{cls}} = \{(x_i, y_i)\}_{i=1}^N$, where x_i is the text and $y_i \in \{\text{'human'}, \text{'ai'}\}$ denotes authorship. Second, **AI Interval Detection** follows $\mathcal{D}_{\text{det}} = \{(x_i, y_i, S_i)\}_{i=1}^M$, where $S_i = \{(start_j, end_j)\}_{j=1}^{K_i}$ is a set of character-level offsets marking the K_i AI-generated spans. S_i is empty for human texts, covers the full text for AI samples, and marks specific regions in mixed texts. Both datasets are partitioned into standard training, validation, and test sets for the experiments in Section 5.

3.2 Classification Dataset Curation

The curation of our classification dataset (Figure 1a) aimed to create a comprehensive and challenging resource. The design was guided by three core principles: diversity across domains and lengths, a wide range of generator models (including both modern and legacy), and complex generation scenarios designed to produce subtle and sophisticated examples.

Human Corpus Collection. The foundation is

a large, diverse corpus of human-authored EN/RU texts aggregated from open-source collections (full list in Appendix B). It spans eight shared domains: **Short-form text, News, Question, Review, Factual text, Poetry, Story, Article**, plus a ninth, **Paper Abstract**, for English to cover academic writing.

Length-Balanced Sampling. To ensure structural balance, we sampled human texts uniformly from predefined word-count buckets. This balance was then propagated to the AI corpus by including explicit output-length instructions within our generation prompts. This process ensures the final AI collection mirrors the length distribution of the human corpus, minimizing the risk of models learning to use text length as a simple heuristic for classification.

AI Text Generation Scenarios. Using the length-balanced and domain-balanced human corpus as a start, we generated a parallel AI corpus with a diverse library of prompt templates. These templates covered four main scenarios (see Appendix D for examples): generating new text based on a human source (**Create**), shortening it via summarization or simplifying (**Delete**), expanding it with more detail (**Expand**), and modifying **outputs from these pipelines** through different methods, e.g., humanization or stylization (**Update**). A key benefit of this process is that every AI text is thematically paired with a human text. This design is crucial as it compels detection models to learn subtle stylistic and structural differences, rather than relying on spurious correlations with the topic.

Generator Model Diversity. To ensure our dataset is not overfitted to the artifacts of a single model family, we utilized a wide spectrum of LLMs. This included: (a) modern proprietary models (e.g., from the Gemini (Comanici et al., 2025) and OpenAI GPT-4 (OpenAI, 2023) families); (b) a diverse set of modern open-source models (e.g., Qwen3 (Yang et al., 2025), DeepSeek-R1 (Guo et al., 2025)); (c) widely-used legacy models (e.g.,

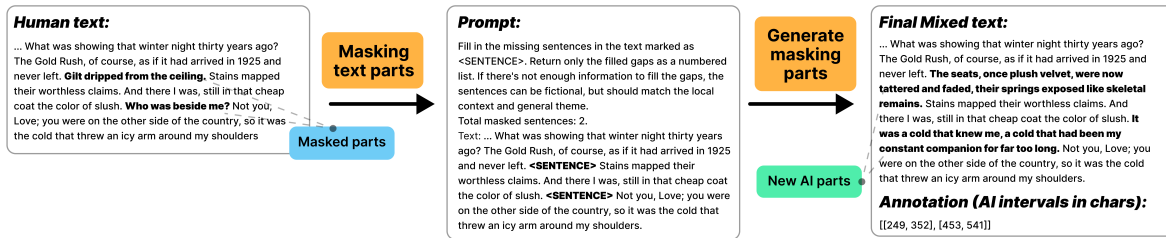


Figure 2: The automated mixed-text generation process: sentences in a human text are masked and then filled in by an LLM, resulting in a text with annotated AI-generated intervals.

GPT-3.5 (Ouyang et al., 2022)); and (d) for the Russian dataset, language-specific models such as GigaChat (Mamedov et al., 2025) and YaGPT¹. We intentionally included models of various sizes (from 760M to 72B parameters) and capabilities (with and without advanced reasoning) to capture a broad range of potential generation signatures. A complete list of all used LLMs is provided in Appendix C.

3.3 Detection Dataset Curation

The detection dataset was curated to provide realistic human-AI collaboration examples with a variable number of AI intervals via three distinct pathways (Figure 1b).

First, **Human Text Gap-Filling** involved masking randomly selected sentences at arbitrary positions within human texts for LLMs to fill (Figure 2). Due to the complexity of this task, we exclusively employed SOTA models (Gemini-2.5-flash (Comanici et al., 2025), GPT-4/4o/o1 (OpenAI, 2023; Hurst et al., 2024; Jaech et al., 2024)) to ensure the highest quality of generated content.

Second, **Human Text Continuation** adapted "continue" prompts (a variant of *Create*) from our classification set, creating single-interval texts that inherit full generator diversity.

Finally, **Manual Editing of AI Texts** produced a unique sub-corpus of the most challenging examples. Though smaller due to the expensive nature of this process, these examples are vital as human modifications break consistent AI statistical patterns, forcing models to learn more nuanced and robust features. Crucially, editors focused solely on content modification: they inserted separator tokens only as a convenience marker within the editing interface to delimit working regions, *not* to define final segment boundaries. Ground-truth labels were computed deterministically via

character-level alignment of the final edited text against the original AI source, making the resulting interval annotations fully automatic and independent of any subjective human judgment (details in Appendix K).

3.4 Post-processing and Filtering

To ensure data quality, all human and AI texts underwent a rigorous filtering pipeline that involved strict language filtering (English or Russian), removing duplicates, low-quality content (e.g., incomplete or repetitive), and any text shorter than five words, including short AI-generated continuations. We also stripped LLM-specific artifacts by filtering common refusal phrases (e.g., "As a language model...") and removing Markdown formatting to normalize the data.

As detailed in Figure 3, the resulting classification datasets comprise ~249k English and ~340k Russian samples (~60/40 AI-to-human ratio), while the detection datasets feature a three-way split of human, AI, and mixed texts. Text-length coverage is broad across all word-count ranges (0 to 500+ words; Appendix Figure 12), and domain representation is robust, averaging ~16.8k-25.2k AI texts for classification and over 1.2k AI and 1.6k mixed texts for detection per domain (see Appendix E). Figure 4 further shows that the average number of AI intervals correlates positively with word count, ensuring the dataset contains challenging examples with multiple, non-trivial AI insertions.

4 Quality and Complexity Assessment

We analyze dataset quality using a dual approach: following Gritsai et al. (2024) to assess deep structural properties via topological and perturbation metrics, and complementing this with textual similarity metrics from the MultiSocial benchmark (Macko et al., 2024) to quantify how closely the output of each LLM resembles the source text it

¹<https://ya.ru/ai/gpt>

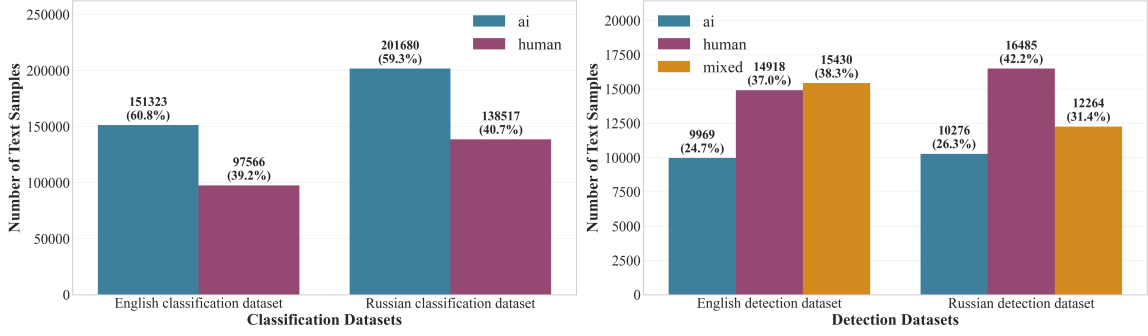


Figure 3: The number of samples for each label in the LLMTrace dataset.

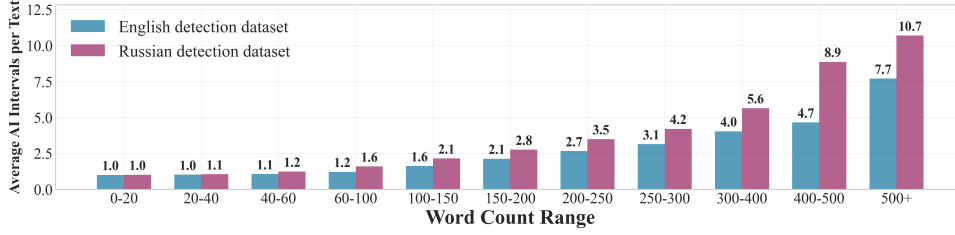


Figure 4: Average number of AI intervals per text in each word count range for the detection datasets.

was based on. These analyses collectively demonstrate how indistinguishable our AI-generated texts are from their human counterparts.

4.1 Topological Statistics

Recent work has shown that the intrinsic dimensionality of a text’s embedding space can serve as a feature to distinguish between human and machine origins [Tulchinskii et al. \(2023\)](#). This is often measured using the Persistence Homology Dimension (PHD). While earlier models tended to produce texts with lower PHD than humans, modern LLMs generate text with much more similar topological complexity.

To quantify the similarity between the PHD distributions of human (h_d) and AI-generated (m_d) texts, we use the symmetric KL-divergence score, KL_{TTS} , as proposed by [Gritsai et al. \(2024\)](#):

$$KL_{TTS}(h_d, m_d) = |D_{KL}(h_d||m_d) - D_{KL}(m_d||h_d)|$$

A lower KL_{TTS} score indicates that the two distributions are more similar, suggesting that the AI texts are harder to distinguish from human texts based on this topological feature. Following [Gritsai et al. \(2024\)](#), we extract embeddings using roberta-base². We acknowledge that using an English-centric encoder (roberta-base) for Russian texts may introduce cross-lingual bias. However, we retained this choice to maintain direct

comparability with the baseline values established in prior work [Gritsai et al. \(2024\)](#) and to isolate dataset-level differences from metric-level variations.

Our English classification dataset achieves a competitive KL_{TTS} of 0.0189 when benchmarked against other public datasets reported by [Gritsai et al. \(2024\)](#) (see Table 2). Notably, our Russian dataset shows an exceptionally low KL_{TTS} of 0.0032, one of the lowest among all compared Russian-language datasets, demonstrating the high quality of our generation pipeline.

The analysis of the detection datasets in Table 3 provides further insights. For both English and Russian, the KL_{TTS} scores between *AI* and *human* texts remain low, confirming the quality of the purely AI-generated texts in this subset. Most importantly, the score between *mixed* and *human* texts is exceptionally low (0.0017 for EN, 0.0332 for RU). This quantitatively confirms that our methods for creating mixed texts produce challenging examples that are topologically very similar to purely human-written content. This conclusion is further supported by the closely aligned mean PHD values shown in Tables 2 and 3, especially the near-identical means for human and mixed texts in the English detection dataset. A visual representation of these highly overlapping distributions is provided in Appendix F.

²<https://huggingface.co/FacebookAI/roberta-base>

Table 2: Quality statistics of our classification datasets vs. public datasets (other values from (Gritsai et al., 2024)); lower KL_{TTS} and $|\Delta_{shift}|$ are better.

Dataset	$KL_{TTS} \downarrow$	PHD_{human}	$PHD_{machine}$	$ \Delta_{shift} \downarrow$
GhostBusterVerma et al. (2023)	0.053	9.84 ± 1.18	9.76 ± 1.15	0.024
MGTBenchHe et al. (2024)	0.043	8.77 ± 1.31	9.97 ± 1.02	0.031
M4Wang et al. (2024b)	0.036	7.26 ± 1.99	8.59 ± 1.40	0.107
MAGELi et al. (2023)	0.011	9.80 ± 2.14	9.38 ± 3.04	0.094
SemEval24 MultiWang et al. (2024a)	0.001	9.65 ± 1.81	9.42 ± 1.44	0.059
RuATDSHamardina et al. (2022)	0.007	7.33 ± 1.40	7.46 ± 1.41	0.315
LLMTrace (EN)	0.0189	9.19 ± 1.87	9.91 ± 1.64	-0.00317
LLMTrace (RU)	0.0032	6.98 ± 0.82	7.06 ± 0.69	0.00074

Table 3: Topological statistics for our detection datasets. The very low KL_{TTS} between human and mixed texts (especially EN) reflects the complexity of mixed-authorship examples.

Dataset	PHD_{human}	PHD_{mixed}	PHD_{ai}	$KL_{TTS} \downarrow$ (mixed vs human)	$KL_{TTS} \downarrow$ (ai vs human)
LLMTrace (EN)	9.27 ± 1.87	9.26 ± 1.83	9.83 ± 1.65	0.0017	0.0233
LLMTrace (RU)	7.12 ± 0.87	7.15 ± 0.63	7.27 ± 0.58	0.0332	0.0483

Table 4: Global average similarity metrics for the English and Russian classification datasets.

Language	METEOR $\uparrow$	BERTScore $\uparrow$	n-gram $\uparrow$	ED-norm (LD) $\downarrow$	LangCheck $\downarrow$	MAUVE* $\downarrow$
LLMTrace (EN)	0.2665	0.6964	0.2164	3.0255	0.0019	0.1713
LLMTrace (RU)	0.1797	0.6767	0.1630	2.4792	0.0045	0.1696

*MAUVE scores were computed on a random sample of 1k (human, AI) pairs per model. Other metrics were calculated on full data.

4.2 Perturbation Statistics

This metric assesses dataset quality by measuring how text embeddings shift after minor, meaning-preserving adversarial changes, measured by the Δ_{shift} score from Gritsai et al. (2024). The perturbation is performed by replacing tokens with synonyms (using WordNet (Miller, 1995) for English and RuWordNet³ for Russian). The multilingual-e5-large⁴ encoder is used to generate embeddings. We report the raw Δ_{shift} values to indicate the directionality of the shift. However, for quality assessment, a value with a magnitude ($|\Delta_{shift}|$) close to zero is desirable, as it signifies that both human and AI texts react similarly to the perturbations.

As shown in Table 2, our English classification dataset achieves a near-zero Δ_{shift} of -0.00317, indicating a very high degree of similarity to human texts in terms of adversarial robustness and outperforming many existing datasets. A visual comparison of the embedding shift distributions for our classification datasets is provided in Appendix G.

4.3 Textual Similarity Metrics

While the preceding metrics assess structural integrity, we also evaluate the direct textual fidelity between AI-generated texts and their human source counterparts. Table 4 summarizes the global average scores for these metrics on our English and Russian classification datasets. We present metrics where higher values indicate greater similarity (METEOR (Banerjee and Lavie, 2005), BERTScore (Zhang et al., 2019), n-gram) and metrics where lower values signify closer resemblance (Levenshtein Distance (LD), LangCheck, MAUVE (Pillutla et al., 2021)). The MAUVE metric is marked with an asterisk (*) to denote that it was calculated on a random sample of 1k text pairs per model. Detailed metric descriptions, as well as per-model and per-prompt type results, are provided in Appendix H.

Together, low KL_{TTS} and near-zero $|\Delta_{shift}|$ show our generated texts are structurally indistinguishable from human writing. Lexical fidelity is equally strong: English BERTScore (0.6964), METEOR (0.2665), and n-gram (0.2164) all surpass top models in MultiSocial (Macko et al., 2024).

³<https://pypi.org/project/ruwordnet/>

⁴<https://huggingface.co/intfloat/multilingual-e5-large>

5 Experiments

To demonstrate the utility of our dataset, we conduct a series of baseline experiments. While simple token-level baselines exist, recent work suggests they struggle with the long-context coherence of modern LLMs. Therefore, we focus on establishing a strong transformer-based baseline using the GigaCheck (Tolstykh et al., 2026) framework. We fine-tune two backbones: Mistral-7B-v0.3 (Jiang et al., 2023) and the smaller mmBERT-base (Marone et al., 2025) (307M parameters). For binary classification, these are trained as standard classifiers; for the interval localization task, following the GigaCheck methodology, we employ a DN-DAB-DETR (Li et al., 2022) head on top of the frozen backbones. All training hyperparameters are detailed in Appendix I.

For all experiments, we enforced strict grouping constraints during data partitioning: paired human and AI texts derived from the same topic were assigned to the same split to prevent content-based data leakage.

5.1 Baseline Performance

We first evaluate the models on our standard held-out test sets.

For the **binary classification task**, we trained English-only (train/valid/test: 173,511 / 36,949 / 38,429), Russian-only (237,929 / 49,747 / 52,521), and bilingual models. The performance is presented in Table 5. Both architectures achieve high F1 scores (0.96), establishing a strong baseline for future comparisons. A detailed per-domain breakdown is available in Appendix J.

Zero-shot detectors lag dramatically: the best, Fast-DetectGPT (Bao et al., 2024), reaches only AI F1 0.69 overall vs. 0.99 for our fine-tuned baseline; DetectGPT (Mitchell et al., 2023) barely exceeds chance (AUROC 0.58); and Binoculars (Hans et al., 2024) collapses on Russian (AI F1 0.02). This confirms the task requires in-domain supervision (Appendix N).

For the **localizing AI-generated intervals** task, we trained the DN-DAB-DETR models on dedicated splits for English (train/valid/test: 27,766 / 5,536 / 7,015) and Russian (26,545 / 5,907 / 6,573). The quantitative results are summarized in Table 6. We report mean Average Precision (mAP) with IoU thresholds fixed at 0.5 and averaged from 0.5 to 0.95.

Table 5: Binary classification results for the English-only (ENG), Russian-only (RU), and Bilingual (BIL) fine-tuned models.

Model	AI F1	Human F1	Mean Acc.	TPR@ 1%FPR
ENG Mistral-7B	0.986	0.979	0.985	0.979
ENG mmBERT	0.972	0.958	0.969	0.941
RU Mistral-7B	0.986	0.980	0.984	0.978
RU mmBERT	0.970	0.958	0.966	0.924
BIL Mistral-7B	0.986	0.980	0.985	0.979
BIL mmBERT	0.969	0.955	0.966	0.930

Table 6: AI interval detection results for the GigaCheck (DN-DAB-DETR) model with frozen Mistral-7B or mmBERT backbones.

Model	mAP@0.5	mAP@0.5:0.95
ENG Mistral-7B	0.8749	0.7555
ENG mmBERT	0.8645	0.7494
RU Mistral-7B	0.8928	0.7839
RU mmBERT	0.8758	0.7629
BIL Mistral-7B	0.8976	0.7921
BIL mmBERT	0.8867	0.7864

5.2 Generalization to Unseen Generators

To assess cross-architecture generalization, we performed **Leave-One-Family-Out (LOFO)** experiments: for each run we removed an entire provider family (all variants and sizes) from training and evaluated a bilingual mmBERT-base model on the held-out family’s target generator. We held out three families chosen as independent architectural lineages spanning the full size range of our open-source pool: **Mistral** (target *Ministral-8B*), **DeepSeek** (target *DeepSeek-R1-Distill-Qwen-32B*), and **Llama** (target *Llama-3.3-70B*). Removing the whole family, not just the target, prevents same-family variants from leaking signatures into training. Table 7 shows negligible drops (< 1.5% AI F1) vs. the fully-trained baseline, indicating that the dataset’s diversity enables learning universal AI markers rather than family-specific artifacts. A complementary population-level analysis across all 14 families in the test set (Section 6; Appendix O.2) shows the dominant difficulty axis is generator scale, not family identity.

5.3 Detection Performance by Generation Strategy

To verify the complexity of different collaboration scenarios, we evaluated detection baselines across mixed-text subsets (Table 8). While models achieve high accuracy on *Gap-Filling* and *Continuation*, performance drops significantly on *Manual Editing*. This confirms that subtle human edits dis-

rupt statistical patterns, validating the importance of including manually edited examples to train robust models capable of detecting sophisticated, “humanized” AI content.

Table 7: Leave-One-Family-Out (LOFO) generalization. Each row holds out the *entire target family* (all variants and sizes) from training and evaluates on its target generator (Human + target only); **Baseline** = full training. Drops are negligible.

Held-out family (target)	Config	AI F1	Human F1	Mean Acc	TPR@1%FPR
DeepSeek	Baseline	0.839	0.987	0.968	0.941
(DeepSeek-R1-Distill, 32B)	LOFO	0.828	0.986	0.968	0.947
Mistral	Baseline	0.811	0.985	0.938	0.859
(Mistral-8B-Instruct, 8B)	LOFO	0.809	0.985	0.935	0.846
Llama	Baseline	0.872	0.987	0.968	0.934
(Llama-3.3-Instruct, 70B)	LOFO	0.877	0.987	0.966	0.932

Table 8: Ablation study of detection performance across different mixed-authorship text generation strategies. The **Manual Editing** strategy proves to be the most challenging for both architectures.

Mixing Strategy	BIL mmBERT		BIL Mistral7B	
	mAP@0.5	mAP@0.5:0.95	mAP@0.5	mAP@0.5:0.95
Gap-Filling	0.826	0.654	0.838	0.660
Continuation	0.788	0.613	0.855	0.692
Manual Editing	0.656	0.458	0.733	0.532

6 Error Analysis

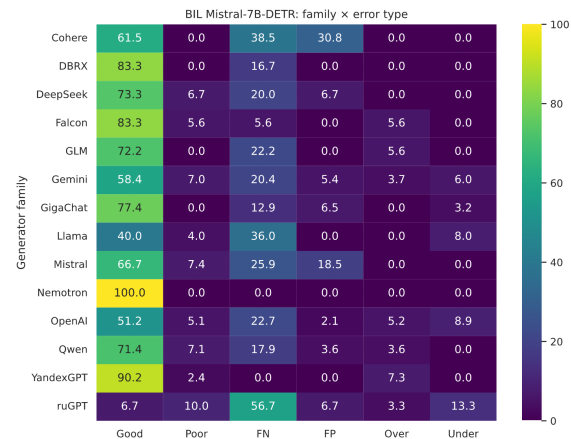


Figure 5: Per-family error-type distribution for BIL Mistral-7B-DETR (% of within-family GT intervals per category). ruGPT and early Llama are hardest; OpenAI/Gemini are easiest. BIL mmBERT-DETR analysis: Appendix Figure 17.

We categorize each predicted/GT interval pair into six error types: *good* and *poor matches* (1-to-1, separated by an IoU threshold of 0.8), *false negatives*, *false positives*, *over-* and *under-segmentation* (definitions in Appendix O). Manual editing yields substantially fewer good matches than the fill-gaps regime (48%/56% for BIL Mistral-7B-DETR;

36%/60% for BIL mmBERT-DETR), confirming that human edits disrupt detectable patterns (Appendix Figure 16).

Slicing errors by generator metadata, the dominant failure mode is on under-represented small open-source models ($\leq 8B$; llama-7b, ruGPT), which reach interval-level F1 of 0.29 - 0.50 vs. 0.83 - 0.90 for frontier closed-source generators (Figure 5). The apparent positive recency trend ($+0.11$ F1/year, $p = 2.4 \times 10^{-4}$ for BIL Mistral-7B-DETR) is an artifact of this same confound: the oldest generators in our test set are also the smallest open-source ones (Appendix O.4). Text length is the strongest text-side factor: recall drops from 0.84 (≤ 200 words) to 0.15 (≥ 2000 words) while precision stays ~ 0.95 , so long contexts produce silent misses rather than false positives. The two detector backbones agree only moderately on per-generator F1 (Spearman $\rho = 0.40$, $p = 0.04$, $n = 28$), indicating that detection difficulty has both data-intrinsic and architecture-specific components. Zero-shot evaluation on RoFT (Dugan et al., 2023) and RoFT-ChatGPT (Kushnareva et al., 2023) (Appendix O.8) reinforces this: transfer is stronger to generators overlapping LLMTrace’s training distribution (GPT-3.5) and weaker to unseen architectures (GPT-2, CTRL). Full breakdowns by family, size, recency, Arena Elo proxy, text properties, and cross-detector statistics are in Appendix O.

7 Conclusion

We presented LLMTrace, a large-scale benchmark addressing critical gaps in detecting modern LLMs and mixed-authorship content. Instantiated for English and Russian via an extensible, language-agnostic curation pipeline, and spanning 38 diverse generators across nine domains, it supports both binary classification and precise interval detection, with extensive analysis confirming the high quality and structural realism of the generated texts. By releasing this resource and establishing strong baselines, we aim to facilitate the development of next-generation detectors capable of handling nuanced, real-world AI usage.

Limitations

While LLMTrace establishes a comprehensive benchmark, we acknowledge specific aspects that outline directions for future research.

First, while our Leave-One-Family-Out (LOFO)

experiments (Section 5.2) hold out three entire provider families (Mistral, DeepSeek, Llama) spanning the full parameter range, and are complemented by a population-level per-family error analysis across all 14 families in the test set (Appendix O.2), running the controlled LOFO protocol exhaustively across every family represented among our 38 generators would provide a fuller picture and remains a valuable direction. Second, regarding evaluation, we established a robust baseline using a detection architecture (DN-DAB-DETR) and standard mAP metrics. Future work could broaden this by benchmarking against sequence-labeling methods and exploring richer span-level metrics (e.g., boundary distance errors) to capture finer localization nuances. Third, the released dataset currently instantiates the curation pipeline for English and Russian; while the pipeline is language-agnostic by design, extending coverage to typologically distinct languages (e.g., CJK, Arabic, Hindi) and to additional language-specific generator families is a natural next step that would further validate the framework’s extensibility. Finally, the significant performance drop on manually edited texts indicates that identifying subtle human modifications remains an open challenge, and the error analysis (Section 6) further shows that detection quality on legacy and under-represented small open-source generators is markedly lower than on frontier closed-source models; broadening the training distribution towards such generators is a clear next step.

Ethical Considerations

Human Labor. All editors who contributed to the mixed-text corpus participated voluntarily, were fully informed of the task, and were compensated at rates significantly above local minimum wage.

Data Privacy. Human-authored texts were sourced from established open-source collections. Standard automated filtering was applied, but incidental PII may remain in web-crawled data; researchers should exercise caution when analyzing individual samples.

Dual-Use Risks. Detection datasets could theoretically help adversaries train less detectable generators, and imperfect detectors risk false positives in academic or professional settings. We believe the benefit of open, high-quality resources for studying these challenges outweighs the risks of withholding them.

References

- Ekaterina Artemova, Jason Lucas, Saranya Venkatesan, Jooyoung Lee, Sergei Tilga, Adaku Uchendu, and Vladislav Mikhailov. 2024. Beemo: Benchmark of expert-edited machine-generated outputs. *arXiv preprint arXiv:2411.04032*.
- Navid Ayoobi, Lily Knab, Wen Cheng, David Pan-toja, Hamidreza Alikhani, Sylvain Flamant, Jin Kim, and Arjun Mukherjee. 2024. Esperanto: Evaluating synthesized phrases to enhance robustness in ai detection for text origination. *arXiv preprint arXiv:2409.14285*.
- 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*.
- Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.
- Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In *International Conference on Learning Representations*, volume 2024, pages 24814–24836.
- Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.
- Hong Chen, Hiroya Takamura, and Hideki Nakayama. 2021. Scixgen: A scientific paper dataset for context-aware text generation. *arXiv preprint arXiv:2110.10774*.
- Gheorghe Comanici, Eric Bieber, Mike Schaeckermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- 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.
- Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. Raid: A shared benchmark for robust evaluation of machine-generated text detectors. *arXiv preprint arXiv:2405.07940*.

- Liam Dugan, Daphne Ippolito, Arun Kirubakaran, 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.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.
- Lukas Gehring and Benjamin Paaßen. 2025. Assessing llm text detection in educational contexts: Does human contribution affect detection? *arXiv preprint arXiv:2508.08096*.
- German Gritsai, Anastasia Voznyuk, Andrey Grabovoy, and Yury Chekhovich. 2024. Are ai detectors good enough? a survey on quality of datasets with machine-generated texts. *arXiv preprint arXiv:2410.14677*.
- 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*.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shitong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: Zero-shot detection of machine-generated text. *arXiv preprint arXiv:2401.12070*.
- 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.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.
- Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.
- Maurice Jakesch, Jeffrey T Hancock, and Mor Naaman. 2023. Human heuristics for ai-generated language are flawed. *Proceedings of the National Academy of Sciences*, 120(11):e2208839120.
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.
- 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.
- Laida Kushnareva, Tatiana Gaintseva, German Magai, Serguei Barannikov, Dmitry Abulkhanov, Kristian Kuznetsov, Eduard Tulchinskii, Irina Piontkovskaya, and Sergey Nikolenko. 2023. Ai-generated text boundary detection with roft. *arXiv preprint arXiv:2311.08349*.
- Mina Lee, Percy Liang, and Qian Yang. 2022. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–19.
- Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. 2022. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13619–13627.
- Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023. Mage: Machine-generated text detection in the wild. *arXiv preprint arXiv:2305.13242*.
- Dominik Macko, Jakub Kopal, Robert Moro, and Ivan Srba. 2024. Multisocial: Multilingual benchmark of machine-generated text detection of social-media texts. *arXiv preprint arXiv:2406.12549*.
- Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, and 1 others. 2023. Multitude: Large-scale multilingual machine-generated text detection benchmark. *arXiv preprint arXiv:2310.13606*.
- Valentin Mamedov, Evgenii Kosarev, Gregory Leleytner, Ilya Shchuckin, Valeriy Berezovskiy, Daniil Smirnov, Dmitry Kozlov, Sergei Averkiev, Lukyanenko Ivan, Aleksandr Proshunin, and 1 others. 2025. Gigachat family: Efficient russian language modeling through mixture of experts architecture. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 93–106.
- Marc Marone, Orion Weller, William Fleshman, Eugene Yang, Dawn Lawrie, and Benjamin Van Durme. 2025. mmbert: A modern multilingual encoder with annealed language learning. *arXiv preprint arXiv:2509.06888*.

- Jiří Milička, Anna Marklová, Ondřej Drobil, and Eva Pospíšilová. 2025. Humans can learn to detect ai-generated texts, or at least learn when they can't. *arXiv preprint arXiv:2505.01877*.
- George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.
- 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.
- Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849.
- Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.
- OpenAI. 2023. [Gpt-4 technical report](#).
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.
- Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.
- Gerrit Quaremba, Elizabeth Black, Denny Vrandečić, and Elena Simperl. 2025. Wetbench: A benchmark for detecting task-specific machine-generated text on wikipedia. *arXiv preprint arXiv:2507.03373*.
- Tatiana Shamardina, Vladislav Mikhailov, Daniil Chernianskii, Alena Fenogenova, Marat Saidov, Anas-tasiya Valeeva, Tatiana Shavrina, Ivan Smurov, Elena Tutubalina, and Ekaterina Artemova. 2022. Findings of the the ruatd shared task 2022 on artificial text detection in russian. *arXiv preprint arXiv:2206.01583*.
- Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th international conference on world wide web*, pages 613–624.
- Irina Tolstykh, Aleksandra Tsybina, Sergey Yakubson, Aleksandr Gordeev, Vladimir Dokholyan, and Maksim Kuprashevich. 2026. [GigaCheck: Detecting LLM-generated content via object-centric span localization](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 4349–4364, San Diego, California, United States. Association for Computational Linguistics.
- Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Sergey Nikolenko, Evgeny Burnaev, Serguei Barannikov, and Irina Piontkovskaya. 2023. Intrinsic dimension estimation for robust detection of ai-generated texts. *Advances in Neural Information Processing Systems*, 36:39257–39276.
- Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2023. Ghostbuster: Detecting text ghost-written by large language models. *arXiv preprint arXiv:2305.15047*.
- Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, and 1 others. 2024a. Semeval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection. *arXiv preprint arXiv:2404.14183*.
- Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, and 1 others. 2024b. M4gt-bench: Evaluation benchmark for black-box machine-generated text detection. *arXiv preprint arXiv:2402.11175*.
- 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.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Sungduk Yu, Man Luo, Avinash Madusu, Vasudev Lal, and Phillip Howard. 2025. Is your paper being reviewed by an llm? benchmarking ai text detection in peer review. *arXiv preprint arXiv:2502.19614*.
- Zijie Zeng, Lele Sha, Yuheng Li, Kaixun Yang, Dragan Gašević, and Guangliang Chen. 2024. Towards automatic boundary detection for human-ai collaborative hybrid essay in education. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22502–22510.
- Qihui Zhang, Chujie Gao, Dongping Chen, Yue Huang, Yixin Huang, Zhenyang Sun, Shilin Zhang, Weiye Li, Zhengyan Fu, Yao Wan, and 1 others. 2024. Llm-as-a-coauthor: Can mixed human-written and

machine-generated text be detected? *arXiv preprint arXiv:2401.05952*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Dmitry Zmitrovich, Alexander Abramov, Andrey Kalmykov, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Vitalii Kadulin, Sergey Markov, and 1 others. 2023. A family of pretrained transformer language models for russian. *arXiv preprint arXiv:2309.10931*.

A Classification Benchmarks Comparison

Table 9 provides a detailed comparison of LLMTrace against 17 existing AI text classification benchmarks along the axes of dataset size, domain coverage, language coverage, number of generators, and prompt-type diversity. This table complements Table 1 in the main text, which focuses on detection (interval-localization) benchmarks.

Table 9: Statistics of the texts in the classification datasets.

Dataset	Dataset Size	# Domains	# Lang	# Gen	Various prompt types
MAGE (Li et al., 2023)	447.7k	7	1	27	✓
MGTBench (He et al., 2024)	21k	3	1	6	✗
BUST (Cornelius et al., 2024)	25.2k	4	1	7	✓
RAID (Dugan et al., 2024)	6.2M	8	1	11	✗
DetectRL (Wu et al., 2024)	235.2k	4	1	4	✓
HC3 (Guo et al., 2023)	125.2k	1	2	1	✗
MULTITuDE (Macko et al., 2023)	74.1k	1	11	8	✗
M4GT-Bench (Wang et al., 2024b)	152.7k	8	8	7	✓
MultiSocial (Macko et al., 2024)	472k	1	22	7	✗
RuATD (Shamardina et al., 2022)	215k	6	1	13	✓
WETBench (Quaremba et al., 2025)	101.9k	1	3	4	✓
Peer Review Detection (Yu et al., 2025)	789k	1	1	5	✗
GEDE (Gehring and Paaßen, 2025)	13.4k	1	1	2	✓
ESPERANTO (Ayoobi et al., 2024)	720k	4	1	8	✓
SHIELD (Ayoobi et al., 2025)	700k	7	1	7	✗
Beemo (Artemova et al., 2024)	19.6k	several	1	10	✓
MixSet (Zhang et al., 2024)	3.6k	6	1	2	✓
LLMTrace (ours)	589,086	9	2	38	✓

B Data Sources

The tables 10 and 11 list the open-source datasets and public sources used to construct the human-authored corpora for the Russian and English languages.

Table 10: Data sources for the Russian corpus.

Dataset Name	Source / Link
SiberianDatasetXL	https://huggingface.co/datasets/SiberiaSoft/SiberianDatasetXL
medical_qa_ru_data	https://huggingface.co/datasets/blinoff/medical_qa_ru_data
mailruQA-big	https://huggingface.co/datasets/Den4ikAI/mailruQA-big
mfaq	https://huggingface.co/datasets/clips/mfaq
miracl-ru-corpus	https://huggingface.co/datasets/Cohere/miracl-ru-corpus-22-12
wiki_lingua	https://huggingface.co/datasets/GEM/wiki_lingua
taiga	https://huggingface.co/datasets/danasone/taiga
IlyaGusev’s datasets	https://huggingface.co/IlyaGusev/datasets
RussianSuperGLUE	https://github.com/RussianNLP/RussianSuperGLUE
xlsum	https://huggingface.co/datasets/csebuetnlp/xlsum
mlsum	https://huggingface.co/datasets/reciTAL/mlsum
xquad	https://huggingface.co/datasets/google/xquad
NestQuad	https://huggingface.co/datasets/NeSTudio/NestQuad
sberquad	https://huggingface.co/datasets/kuznetsoffandrey/sberquad
ru_sentiment_dataset	https://huggingface.co/datasets/MonoHime/ru_sentiment_dataset
restaurants_reviews	https://huggingface.co/datasets/blinoff/restaurants_reviews
cedr	https://huggingface.co/datasets/sagteam/cedr_v1

C Used generation models

The table 12 lists the open-source and proprietary Large Language Models (LLMs) used to construct the AI-written corpora for the Russian and English languages.

Table 11: Data sources for the English corpus.

Dataset Name / Source	Link / Reference
medium-articles	https://www.kaggle.com/datasets/fabiochiusano/medium-articles
wiki-22-12	https://huggingface.co/datasets/Cohere/wikipedia-22-12
cnn-dailymail	https://www.kaggle.com/datasets/gowrishankarp/newspaper-text-summarization-cnn-dailymail
plot-synopses	https://www.kaggle.com/datasets/cryptexcode/mpst-movie-plot-synopses-with-tags
foundation-poems	https://www.kaggle.com/datasets/tgdivy/poetry-foundation-poems
amazon-questions	https://www.kaggle.com/datasets/praneshmukhopadhyay/amazon-questionanswer-dataset
yahoo-answers	https://www.kaggle.com/datasets/yacharki/yahoo-answers-10-categories-for-nlp-csv
amazon-reviews	https://www.kaggle.com/datasets/kritanjali/jain/amazon-reviews
sentiment-tweets	https://www.kaggle.com/datasets/tariqsays/sentiment-dataset-with-1-million-tweets
ask-reddit	https://www.kaggle.com/datasets/gpreda/ask-reddit
arxiv-abstracts	https://www.kaggle.com/datasets/spsayakpaul/arxiv-paper-abstracts
Common Crawl	https://commoncrawl.org/
SciXGen	(Chen et al., 2021)
XSum	(Narayan et al., 2018)
TLDR_news	https://huggingface.co/datasets/JulesBelveze/TLDR_news
Reddit WritingPrompts	(Fan et al., 2018)
ROCStories Corpora	(Mostafazadeh et al., 2016)
Yelp dataset	(Zhang et al., 2015)
/r/ChangeMyView (CMV)	(Tan et al., 2016)
News Outlets	CNN, Washington Post, New York Times
Academic Sources	arXiv (CS, physics), Springer's SSRN (HHS)

Table 12: List of LLMs used for AI text generation.

Model	Languages	Proprietary	Source
<i>Open-Source Models</i>			
AI21-Jamba-Mini-1.5	RU	No	https://huggingface.co/ai21labs/AI21-Jamba-Mini-1.5
c4ai-command-r-08-2024	RU+EN	No	https://huggingface.co/CohereLabs/c4ai-command-r-08-2024
dbx-instruct	RU+EN	No	https://huggingface.co/databricks/dbx-instruct
DeepSeek-R1-Distill-Qwen-32B	RU+EN	No	https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-32B
Falcon3-10B-Instruct	EN	No	https://huggingface.co/tiiuae/Falcon3-10B-Instruct
FRED-T5-1.7B	RU	No	https://huggingface.co/ai-forever/FRED-T5-1.7B
gemma-1.1-7b-it	RU	No	https://huggingface.co/google/gemma-1.1-7b-it
gemma-2-27b-it	RU	No	https://huggingface.co/google/gemma-2-27b-it
GLM-4-32B-0414	EN	No	https://huggingface.co/zai-org/GLM-4-32B-0414
Jamba-v0.1	RU	No	https://huggingface.co/ai21labs/Jamba-v0.1
Llama-3.1-Nemotron-70B-Instruct-HF	EN	No	https://huggingface.co/nvidia/Llama-3.1-Nemotron-70B-Instruct-HF
Llama-3.3-70B-Instruct	RU+EN	No	https://huggingface.co/unsloth/Llama-3.3-70B-Instruct
llama-7b	RU	No	https://huggingface.co/baffo32/decapoda-research-llama-7b-hf
Magistral-Small-2506	EN	No	https://huggingface.co/mistralai/Magistral-Small-2506
Meta-Llama-3-8B-Instruct	RU	No	https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
Ministral-8B-Instruct-2410	RU+EN	No	https://huggingface.co/mistralai/Ministral-8B-Instruct-2410
Phi-3-medium-128k-instruct	RU	No	https://huggingface.co/microsoft/Phi-3-medium-128k-instruct
Phi-3-mini-128k-instruct	RU	No	https://huggingface.co/microsoft/Phi-3-mini-128k-instruct
Qwen2-7B-Instruct	RU	No	https://huggingface.co/Qwen/Qwen2-7B-Instruct
Qwen2.5-72B-Instruct	RU+EN	No	https://huggingface.co/Qwen/Qwen2.5-72B-Instruct
Qwen3-32B	EN	No	https://huggingface.co/Qwen/Qwen3-32B
QwQ-32B	RU	No	https://huggingface.co/Qwen/QwQ-32B
ruGPT family (rugpt3large, rugpt2large)	RU	No	(Zmitrovich et al., 2023)
WizardLM-2-7B	RU	No	https://huggingface.co/dreamgen/WizardLM-2-7B
YandexGPT-5-Lite-8B-instruct	RU	No	https://huggingface.co/yandex/YandexGPT-5-Lite-8B-instruct
Yi-1.5-34B-Chat	RU	No	https://huggingface.co/01-ai/Yi-1.5-34B-Chat
<i>Proprietary Models</i>			
GigaChat series (Pro, Max)	RU	Yes	https://giga.chat/
Google Gemini series (2.0/2.5 flash)	RU+EN	Yes	https://gemini.google.com/
OpenAI GPT series (3.5, 4, 4o, o1, o1-mini, o3)	RU+EN	Yes	https://openai.com/
YaGPT 2	RU	Yes	https://ya.ru/ai/gpt

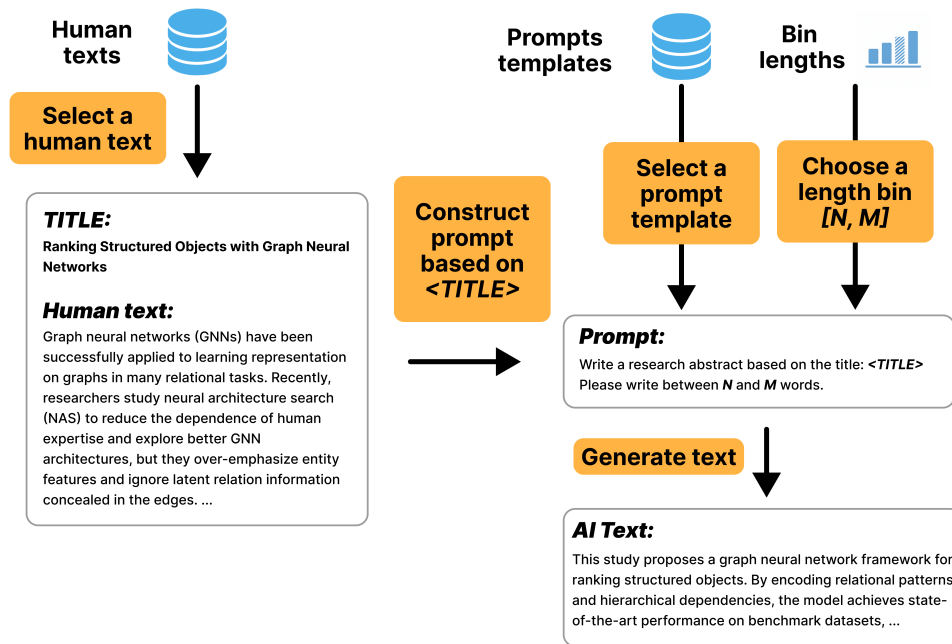


Figure 6: Example of the **Create** generation pipeline, where a research abstract is generated from a title.

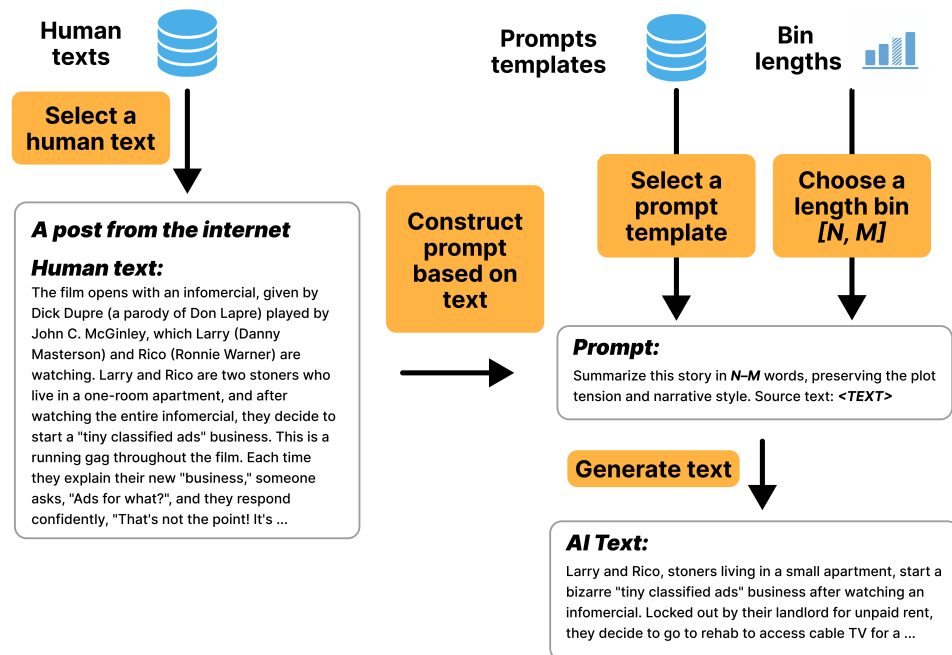


Figure 7: Example of the **Delete** generation pipeline, where a long post is summarized.

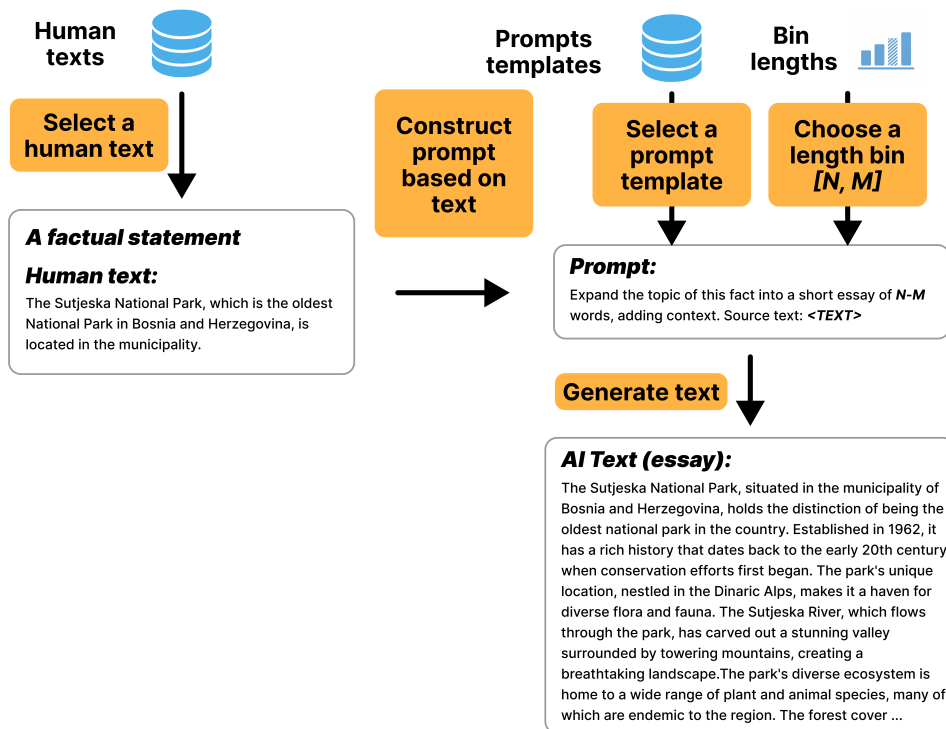


Figure 8: Example of the **Expand** generation pipeline, where a factual statement is expanded into an essay.

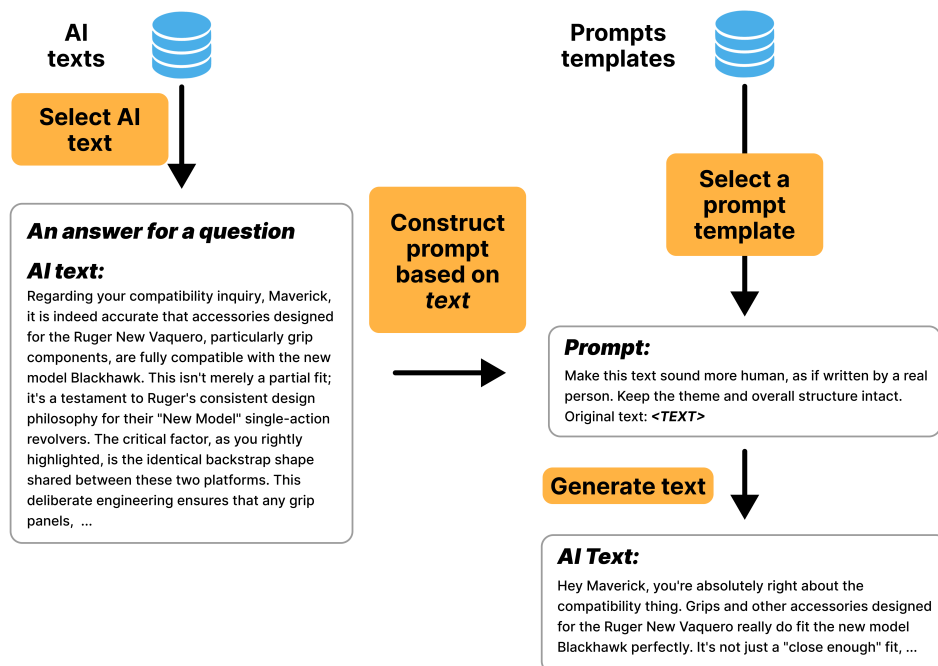


Figure 9: Example of the **Update** generation pipeline, where an AI-generated text is modified to sound more human.

D Prompt Generation Examples

This section provides detailed visual diagrams illustrating the four primary categories of prompt templates used to generate the AI text corpus: *Create*, *Delete*, *Expand*, and *Update*. Each figure demonstrates the end-to-end process, from the selection of a source text to the final AI-generated output.

The **Create** pipeline (Figure 6) uses a specific attribute from a human text, such as its title, to construct a prompt for generating a new, topically-related AI text within a target length bin.

The **Delete** pipeline (Figure 7) takes an entire human text as input and uses a prompt to generate a condensed, summary version of it, again adhering to a length constraint.

The **Expand** pipeline (Figure 8) operates similarly but with the opposite goal: it uses a prompt to elaborate on a short human text, generating a more detailed and verbose AI text.

Finally, the **Update** pipeline, depicted in Figure 9, is unique. It begins with an already-generated AI text and applies a modification prompt, for instance, to "humanize" the writing style or change its tone, to create a more complex and subtle AI artifact. Note that this pipeline does not typically require a length constraint, as the goal is stylistic transformation rather than content resizing.

E Dataset Statistics and Analysis

This section provides a detailed breakdown of the sample distribution across the various domains for both the classification and detection datasets. Figures 10 and 11 illustrates the count of human, AI, and mixed texts within each domain for the English and Russian corpora. These plots visually confirm the broad and balanced coverage of genres, which is a key feature of our dataset designed to enhance the generalizability of detection models.

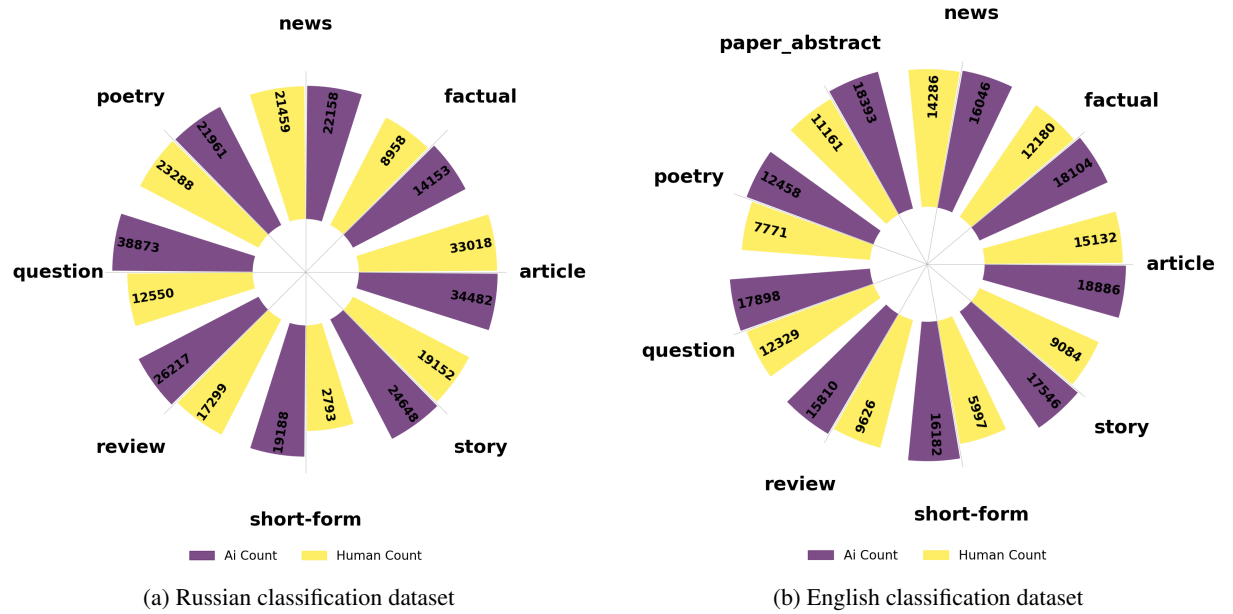
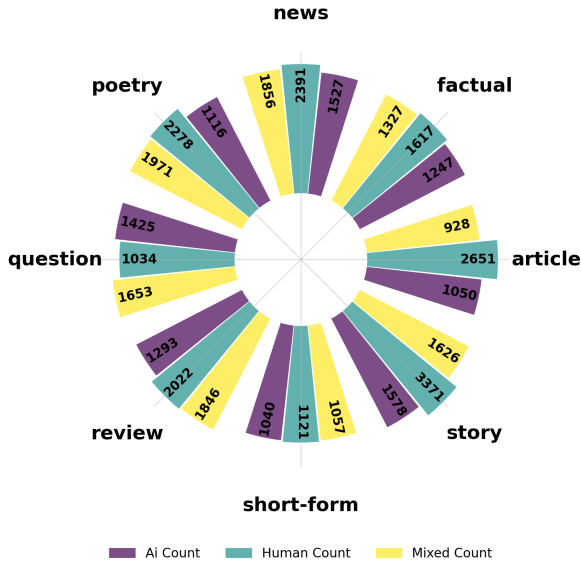


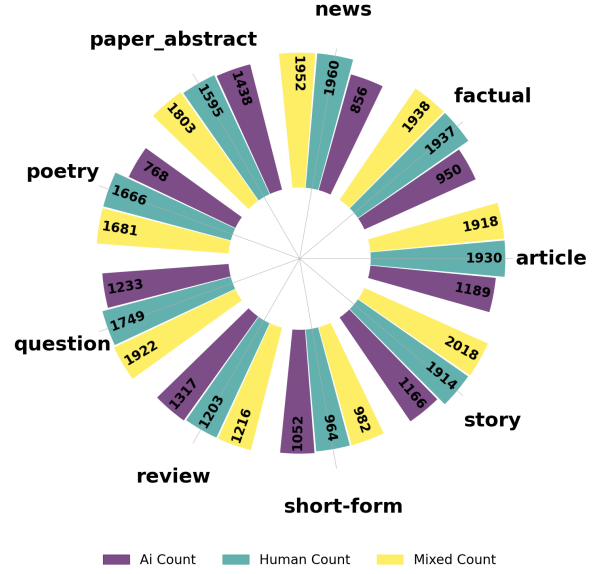
Figure 10: Distribution of labels across domains for classification datasets. Each plot shows the number of samples for each label type within a specific domain, for both Russian and English languages.

F PHD Distributions

Figure 13 provides a visual representation of the Persistence Homology Dimension (PHD) distributions for all categories (human, AI, and mixed) across our classification and detection datasets for both languages. These plots serve as a visual confirmation of the low KL_{TTS} scores and closely aligned mean PHD values reported in Section 4. The significant overlap between the distributions for all text types highlights

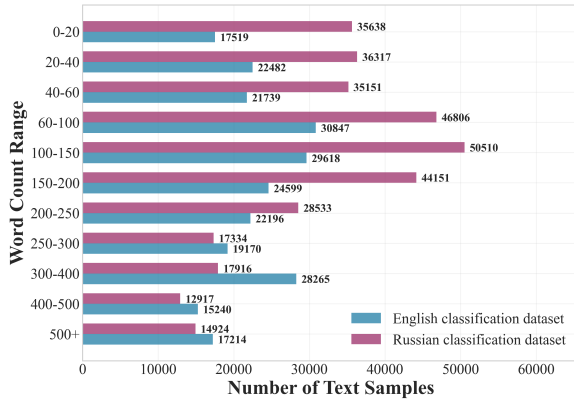


(a) Russian detection dataset

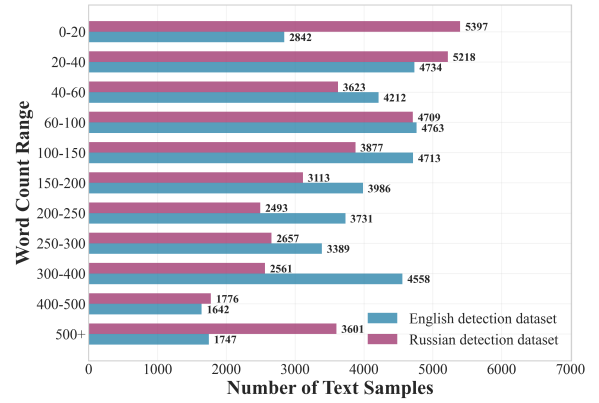


(b) English detection dataset

Figure 11: Distribution of labels across domains for detection datasets. Each plot shows the number of samples for each label type within a specific domain, for both Russian and English languages.



(a) Classification datasets



(b) Detection datasets

Figure 12: The distribution of text samples across different word count ranges for each dataset.

the structural and topological similarity between our AI-generated, mixed, and human-authored texts, underscoring the challenge our dataset presents for detection models.

G Embedding Shift Distributions after Perturbation

This section provides a visual comparison of the embedding shifts for human and AI texts from our classification datasets after undergoing adversarial token perturbation (synonym replacement). The plots in Figure 14 show the distributions of cosine distances between the embeddings of original texts and their perturbed versions.

The close alignment and significant overlap between the distributions for human and AI texts in both the English (Figure 14b) and Russian (Figure 14a) datasets visually confirm the low $|\Delta_{\text{shift}}|$ scores reported in Section 4. This indicates that the AI-generated texts in our dataset exhibit a level of robustness to semantic perturbations that is highly comparable to that of human-authored texts, highlighting the quality and challenge of the dataset.

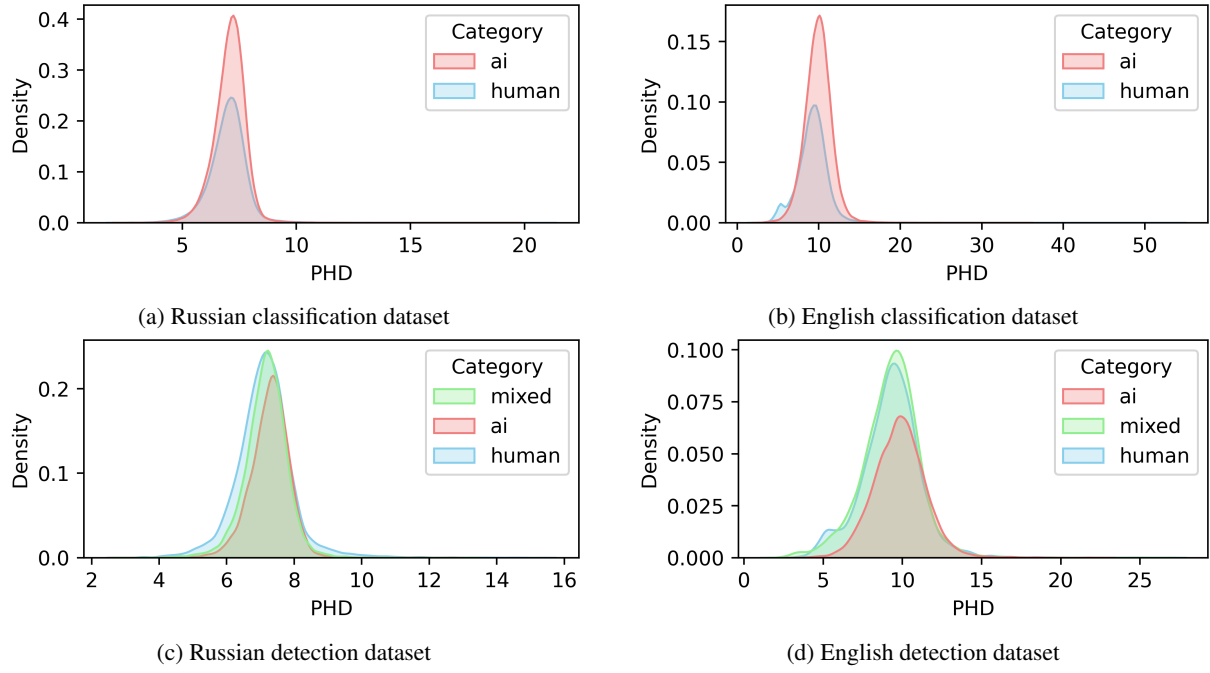


Figure 13: PHD comparison for classification (AI vs. Human) and detection (AI vs. Human vs. Mixed) datasets for Russian and English languages.

H Textual Similarity Metric Descriptions and Detailed Results

H.1 Similarity Metric Descriptions

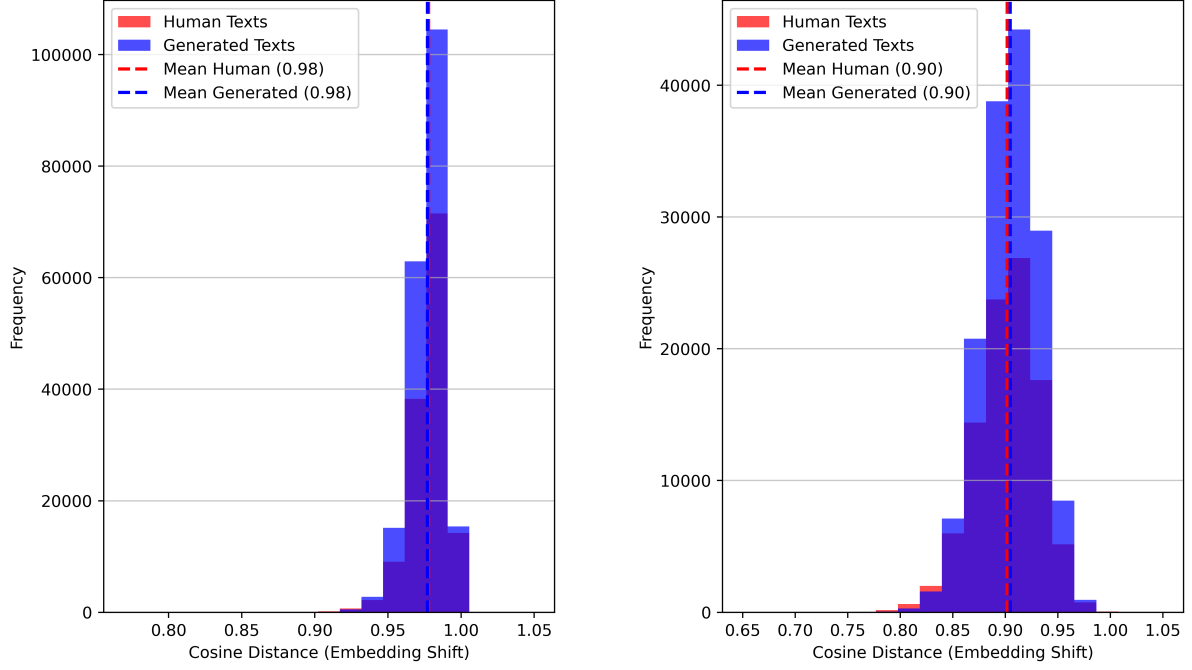
To provide a comprehensive assessment of text similarity, we employ a suite of metrics, each capturing a different aspect of the relationship between the original human text and the machine-generated paraphrase.

- **METEOR** (Banerjee and Lavie, 2005) (Metric for Evaluation of Translation with Explicit ORdering) is a standard metric in machine translation that measures similarity based on unigram alignments between a reference and a hypothesis text, considering precision, recall, and fragmentation.
- **BERTScore** (Zhang et al., 2019) computes similarity by comparing the contextual embeddings of tokens from a reference and a hypothesis text. It uses a multilingual BERT model, making it robust for cross-lingual comparisons and sensitive to semantic similarity beyond exact word matches.
- **n-gram Similarity**⁵ is a language-independent metric that calculates the ratio of shared n-grams (in our case, 3-grams) between two strings, providing a measure of surface-level lexical overlap.
- **Levenshtein Distance (LD)**, reported as **ED-norm**, is a character-level metric that counts the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one string into the other. We normalize this value by the character length of the human text. This calculation is performed using the `editdistance`⁶ library, consistent with the methodology of the MultiSocial benchmark (Macko et al., 2024).
- **LangCheck** measures the percentage of generated texts for which the language differs from the language of the original human text, where both languages are detected using FastText⁷. It serves as an indicator of potential style-imitation artifacts or language inconsistencies.
- **MAUVE** (Pillutla et al., 2021) is a distributional metric that measures the gap between the distribution of machine-generated and human-written texts. It uses clustering on text embeddings, obtained from

⁵<https://pypi.org/project/ngram/>

⁶<https://github.com/roy-ht/editdistance>

⁷<https://fasttext.cc/>



(a) Embedding shifts for the Russian classification dataset.

(b) Embedding shifts for the English classification dataset.

Figure 14: Distributions of cosine distances between original and perturbed text embeddings for human and AI classes. The significant overlap demonstrates similar robustness to synonym-based perturbations for both classes.

the google-bert/bert-base-multilingual-cased model, to quantify the divergence between the two text sets. A lower score indicates that the AI-generated distribution is closer to the human one. Due to its computational intensity, all MAUVE scores were calculated on random data samples. For the per-model analysis (Appendix H.3), our sample consisted of 1,000 pairs per model, for a total of 15,000 English and 26,162 Russian pairs. For the per-prompt type analysis (Appendix H.2), we sampled 3,000 pairs for each of the four prompt types, yielding a total of 12,000 pairs per language.

H.2 Per-Prompt Type Similarity Results

To further analyze the characteristics of our dataset, we also computed the textual similarity metrics grouped by the type of prompt used for generation. This allows us to understand how different types of prompts influence the similarity between the AI-generated output and the human source text. The results for the English and Russian datasets are presented in Table 13 and Table 14, respectively.

Table 13: Per-prompt type similarity metrics for the English dataset. Asterisk (*) denotes that the MAUVE metric was calculated on a random sample of 3,000 text pairs per prompt type.

Prompt Type	METEOR $\uparrow$	BERTScore $\uparrow$	n-gram $\uparrow$	ED-norm $\downarrow$	LangCheck $\downarrow$	MAUVE* $\downarrow$
create	0.3955 (± 0.3010)	0.7457 (± 0.1274)	0.3070 (± 0.2313)	2.1116 (± 5.0455)	0.0023 (± 0.0483)	0.3307
delete	0.1542 (± 0.1177)	0.6967 (± 0.0614)	0.2100 (± 0.1308)	0.7986 (± 0.1554)	0.0008 (± 0.0290)	0.0645
expand	0.3090 (± 0.1447)	0.6877 (± 0.0610)	0.1706 (± 0.1094)	5.5252 (± 9.2193)	0.0025 (± 0.0501)	0.0673
update	0.2022 (± 0.1045)	0.6578 (± 0.0526)	0.1792 (± 0.1017)	3.4772 (± 7.8540)	0.0018 (± 0.0426)	0.1389
Global	0.2665 (± 0.2071)	0.6964 (± 0.0876)	0.2164 (± 0.1618)	3.0255 (± 6.8643)	0.0019 (± 0.0435)	0.1675

Table 14: Per-prompt type similarity metrics for the Russian dataset. Asterisk (*) denotes that the MAUVE metric was calculated on a random sample of 3,000 text pairs per prompt type.

Prompt Type	METEOR $\uparrow$	BERTScore $\uparrow$	n-gram $\uparrow$	ED-norm $\downarrow$	LangCheck $\downarrow$	MAUVE* $\downarrow$
create	0.1616 ($\pm$ 0.1217)	0.6568 ($\pm$ 0.0577)	0.1307 ($\pm$ 0.0884)	2.5010 ($\pm$ 4.6219)	0.0037 ($\pm$ 0.0608)	0.1719
delete	0.1235 ($\pm$ 0.1186)	0.6894 ($\pm$ 0.0602)	0.1684 ($\pm$ 0.1226)	1.2259 ($\pm$ 2.7432)	0.0060 ($\pm$ 0.0770)	0.0829
expand	0.3223 ($\pm$ 0.1780)	0.7053 ($\pm$ 0.0688)	0.2179 ($\pm$ 0.1372)	4.0885 ($\pm$ 21.1010)	0.0068 ($\pm$ 0.0823)	0.1175
update	0.1648 ($\pm$ 0.1303)	0.6786 ($\pm$ 0.0551)	0.1752 ($\pm$ 0.1062)	2.5295 ($\pm$ 7.7285)	0.0028 ($\pm$ 0.0529)	0.2235
Global	0.1797 ($\pm$ 0.1482)	0.6767 ($\pm$ 0.0623)	0.1630 ($\pm$ 0.1135)	2.4792 ($\pm$ 9.7793)	0.0045 ($\pm$ 0.0670)	0.1887

H.3 Per-Model Similarity Results

The following tables provide a detailed breakdown of performance for each LLM, evaluated separately on the English (Table 15) and Russian (Table 16) datasets.

Table 15: Per-model similarity metrics for the English dataset. Asterisk (*) denotes that the MAUVE metric was calculated on a random sample of 1,000 text pairs per model. Other metrics were calculated on the full English classification dataset.

Model	METEOR $\uparrow$	BERTScore $\uparrow$	n-gram $\uparrow$	ED-norm $\downarrow$	LangCheck $\downarrow$	MAUVE* $\downarrow$
c4ai-command-r-08-2024	0.2515 ($\pm$ 0.2003)	0.6903 ($\pm$ 0.0881)	0.2119 ($\pm$ 0.1674)	2.8851 ($\pm$ 7.7371)	0.0017 ($\pm$ 0.0417)	0.1765
Qwen2.5-72B-Instruct	0.2782 ($\pm$ 0.2083)	0.7033 ($\pm$ 0.0849)	0.2248 ($\pm$ 0.1661)	3.0838 ($\pm$ 6.8450)	0.0013 ($\pm$ 0.0360)	0.3581
Qwen3-32B	0.2499 ($\pm$ 0.1859)	0.6866 ($\pm$ 0.0816)	0.2053 ($\pm$ 0.1462)	3.2182 ($\pm$ 6.9384)	0.0023 ($\pm$ 0.0479)	0.1893
dbx-instruct	0.2843 ($\pm$ 0.1877)	0.7043 ($\pm$ 0.0828)	0.2302 ($\pm$ 0.1591)	3.0995 ($\pm$ 6.3418)	0.0039 ($\pm$ 0.0625)	0.2929
DeepSeek-R1-Distill-Qwen-32B	0.2656 ($\pm$ 0.1969)	0.6984 ($\pm$ 0.0833)	0.2183 ($\pm$ 0.1591)	3.0828 ($\pm$ 6.5980)	0.0017 ($\pm$ 0.0412)	0.2872
gemini-2.0-flash	0.2529 ($\pm$ 0.2212)	0.6921 ($\pm$ 0.0930)	0.2105 ($\pm$ 0.1652)	2.7950 ($\pm$ 6.5796)	0.0014 ($\pm$ 0.0379)	0.3915
gemini-2.5-flash	0.2644 ($\pm$ 0.2218)	0.6878 ($\pm$ 0.0907)	0.2064 ($\pm$ 0.1510)	3.3609 ($\pm$ 9.2018)	0.0016 ($\pm$ 0.0399)	0.2379
gpt-4.1-2025-04-14	0.2601 ($\pm$ 0.2041)	0.6975 ($\pm$ 0.0876)	0.2164 ($\pm$ 0.1485)	2.7985 ($\pm$ 5.7855)	0.0012 ($\pm$ 0.0351)	0.2780
Magistral-Small-2507	0.3149 ($\pm$ 0.2476)	0.7242 ($\pm$ 0.1006)	0.2542 ($\pm$ 0.2036)	2.5281 ($\pm$ 5.7650)	0.0026 ($\pm$ 0.0509)	0.4683
Ministral-8B-Instruct-2410	0.2831 ($\pm$ 0.2414)	0.7135 ($\pm$ 0.0945)	0.2329 ($\pm$ 0.1917)	2.2827 ($\pm$ 5.1872)	0.0018 ($\pm$ 0.0423)	0.3413
Llama-3.1-Nemotron-70B-Instruct-HF	0.2575 ($\pm$ 0.1717)	0.6796 ($\pm$ 0.0768)	0.2019 ($\pm$ 0.1356)	3.7079 ($\pm$ 7.5270)	0.0018 ($\pm$ 0.0425)	0.1534
o3-2025-04-16	0.2257 ($\pm$ 0.1816)	0.6798 ($\pm$ 0.0808)	0.1870 ($\pm$ 0.1279)	3.0114 ($\pm$ 6.2277)	0.0022 ($\pm$ 0.0467)	0.2437
Falcon3-10B-Instruct	0.2729 ($\pm$ 0.2160)	0.7033 ($\pm$ 0.0895)	0.2224 ($\pm$ 0.1716)	2.7260 ($\pm$ 5.7081)	0.0013 ($\pm$ 0.0355)	0.3005
Llama-3.3-70B-Instruct	0.2611 ($\pm$ 0.1905)	0.6895 ($\pm$ 0.0835)	0.2107 ($\pm$ 0.1597)	3.4868 ($\pm$ 8.4771)	0.0017 ($\pm$ 0.0408)	0.2358
GLM-4-32B-0414	0.2923 ($\pm$ 0.2193)	0.7065 ($\pm$ 0.0865)	0.2268 ($\pm$ 0.1653)	3.0156 ($\pm$ 5.8937)	0.0022 ($\pm$ 0.0464)	0.3191

I GigaCheck Training Hyperparameters

All models were trained using the transformers⁸ library.

Classification Models (Mistral-7B and mmBERT-base). The classification models were trained with the following key hyperparameters:

⁸<https://github.com/huggingface/transformers>

Table 16: Per-model similarity metrics for the Russian dataset. Asterisk (*) denotes that the MAUVE metric was calculated on a random sample of 1,000 text pairs per model. Other metrics were calculated on the full Russian classification dataset.

Model	METEOR $\uparrow$	BERTScore $\uparrow$	n-gram $\uparrow$	ED-norm $\downarrow$	LangCheck $\downarrow$	MAUVE* $\downarrow$
Yi-1.5-34B-Chat	0.1957 (± 0.1529)	0.6961 (± 0.0596)	0.1965 (± 0.1160)	1.9419 (± 3.5813)	0.0065 (± 0.0804)	0.2268
c4ai-command-r-08-2024	0.1616 (± 0.1074)	0.6787 (± 0.0466)	0.1776 (± 0.0973)	2.2883 (± 4.0365)	0.0014 (± 0.0372)	0.1627
GigaChat-Max	0.1497 (± 0.1215)	0.6781 (± 0.0482)	0.1693 (± 0.0933)	2.1745 (± 3.5091)	0.0010 (± 0.0314)	0.2129
Jamba-v0.1	0.1093 (± 0.1436)	0.6360 (± 0.0716)	0.1035 (± 0.1033)	1.7394 (± 3.0358)	0.0000 (± 0.0000)	0.1391
Meta-Llama-3-8B-Instruct	0.2137 (± 0.1346)	0.6728 (± 0.0640)	0.1695 (± 0.0916)	3.3092 (± 5.8424)	0.0012 (± 0.0346)	0.0539
Phi-3-mini-128k-instruct	0.2120 (± 0.1312)	0.6461 (± 0.0607)	0.1456 (± 0.0808)	3.5791 (± 5.9486)	0.0055 (± 0.0741)	0.0513
QwQ-32B	0.1496 (± 0.1007)	0.6787 (± 0.0448)	0.1640 (± 0.0892)	2.7626 (± 17.3150)	0.0017 (± 0.0413)	0.1776
Qwen2.5-72B-Instruct	0.2117 (± 0.1703)	0.7000 (± 0.0592)	0.1956 (± 0.1245)	2.2224 (± 4.1846)	0.0014 (± 0.0375)	0.3678
Qwen2-7B-Instruct	0.1831 (± 0.1114)	0.6653 (± 0.0585)	0.1536 (± 0.0920)	4.8198 (± 8.1632)	0.0008 (± 0.0275)	0.0523
T5	0.1775 (± 0.1430)	0.6979 (± 0.0642)	0.1388 (± 0.1022)	2.6308 (± 3.3524)	0.0000 (± 0.0000)	0.9973
WizardLM-2-7B	0.2030 (± 0.1338)	0.6676 (± 0.0581)	0.1642 (± 0.0928)	3.1223 (± 5.5773)	0.0043 (± 0.0652)	0.0453
dbx-instruct	0.2167 (± 0.1837)	0.6999 (± 0.0659)	0.1952 (± 0.1366)	2.1221 (± 5.2134)	0.0206 (± 0.1419)	0.2743
DeepSeek-R1-Distill-Qwen-32B	0.1732 (± 0.1578)	0.6879 (± 0.0621)	0.1738 (± 0.1207)	3.2682 (± 40.5175)	0.0223 (± 0.1478)	0.2339
gemma-1.1-7b-it	0.1493 (± 0.1038)	0.6363 (± 0.0542)	0.1278 (± 0.0718)	3.9172 (± 6.7532)	0.0051 (± 0.0714)	0.0082
GigaChat	0.1506 (± 0.1314)	0.6573 (± 0.0663)	0.1272 (± 0.1007)	3.2317 (± 5.1930)	0.0024 (± 0.0492)	0.1378
gemma-2-27b-it	0.1730 (± 0.1381)	0.6810 (± 0.0494)	0.1724 (± 0.1003)	1.9631 (± 3.1443)	0.0015 (± 0.0388)	0.2404
gpt-3.5	0.1766 (± 0.1273)	0.6646 (± 0.0573)	0.1474 (± 0.0934)	3.0603 (± 4.3994)	0.0015 (± 0.0393)	0.1169
gpt-4-0125-preview	0.1076 (± 0.0513)	0.6201 (± 0.0379)	0.0990 (± 0.0647)	5.7928 (± 8.0727)	0.0047 (± 0.0682)	0.0846
gpt-4-1106-preview	0.1053 (± 0.0503)	0.6182 (± 0.0371)	0.0934 (± 0.0619)	5.5213 (± 7.7930)	0.0021 (± 0.0460)	0.0629
gpt-4o	0.1178 (± 0.0795)	0.6455 (± 0.0463)	0.0921 (± 0.0570)	1.0561 (± 0.6252)	0.0060 (± 0.0771)	0.1094
llama-7b	0.2260 (± 0.1361)	0.6798 (± 0.0577)	0.1753 (± 0.0953)	2.7548 (± 4.2883)	0.0018 (± 0.0422)	0.3801
Phi-3-medium-128k-instruct	0.2053 (± 0.1200)	0.7183 (± 0.0558)	0.2652 (± 0.1203)	5.3981 (± 21.8489)	0.0095 (± 0.0976)	0.1389
Ministral-8B-Instruct-2410	0.2689 (± 0.2380)	0.7254 (± 0.0799)	0.2351 (± 0.1815)	1.8615 (± 8.5790)	0.0024 (± 0.0485)	0.3012
o1-mini-2024-09-12	0.1881 (± 0.1108)	0.6748 (± 0.0507)	0.1849 (± 0.0884)	1.5011 (± 2.3402)	0.0000 (± 0.0000)	0.2687
o1-preview-2024-09-12	0.1785 (± 0.1134)	0.6789 (± 0.0540)	0.2330 (± 0.1115)	1.9907 (± 5.2266)	0.0000 (± 0.0000)	0.3393
o3-2025-04-16	0.1523 (± 0.1003)	0.6680 (± 0.0464)	0.1549 (± 0.0849)	3.0001 (± 8.0777)	0.0042 (± 0.0649)	0.1021
ruGPT	0.2053 (± 0.1250)	0.6605 (± 0.0595)	0.1416 (± 0.0901)	5.8582 (± 9.0776)	0.0032 (± 0.0566)	0.2552
Llama-3.3-70B-Instruct	0.2263 (± 0.1788)	0.7000 (± 0.0622)	0.2033 (± 0.1273)	2.4090 (± 5.8166)	0.0012 (± 0.0345)	0.1998
YaGPT	0.1777 (± 0.1371)	0.6646 (± 0.0639)	0.1520 (± 0.0964)	2.6590 (± 3.7624)	0.0006 (± 0.0239)	0.1359
YandexGPT-5-Lite-8B-instruct	0.2241 (± 0.1791)	0.7025 (± 0.0630)	0.2009 (± 0.1269)	2.0329 (± 3.4634)	0.0029 (± 0.0536)	0.2323

- **Pretrained model:** Mistral-7B-v0.3 / mmBERT-base
- **Sequence length:** max 1024, min 100, random sequence length enabled
- **LoRA:** rank=8, alpha=16
- **Precision:** bf16
- **Batch size per GPU:** 64 (train), 1 (eval)
- **Number of GPUs:** 8
- **Gradient accumulation steps:** 1
- **Optimizer:** AdamW
- **Learning rate:** $3 \cdot 10^{-5}$, cosine scheduler with min LR scaled by 0.5
- **Warmup steps:** 20
- **Number of epochs:** 20
- **Random seed:** 8888
- **Classifier pooling for mmBERT-base:** CLS

Detection Model (GigaCheck DN-DAB-DETR). The DN-DAB-DETR detection models were trained with the following key hyperparameters:

- **Feature extractor:** Mistral-7B-v0.3 (frozen) / mmBERT-base (frozen)
- **Sequence length:** max 1024, min 100, random sequence length enabled
- **Precision:** bf16 for the frozen feature extractor, fp32 for the trained DN-DAB-DETR
- **Batch size per GPU:** 64 (train), 1 (eval)
- **Gradient accumulation steps:** 1
- **Number of GPUs:** 8
- **Optimizer:** AdamW
- **Weight decay:** $1 \cdot 10^{-4}$
- **Learning rate:** $2 \cdot 10^{-4}$, cosine scheduler with min LR scaled by 0.5
- **Warmup steps:** 100
- **Number of epochs:** 150
- **DETR parameters:** 45 queries, 3 encoder and decoder layers, input embedding dimension 256

J Detailed Classification Results

To provide a more granular analysis of our binary classification models’ performance, we report the results disaggregated by several key factors. Table 17 shows the performance across different text domains. Table 18 presents the results broken down by text length, grouped into bins based on word count. Finally, Table 19 details the performance for each of the prompt types used during data generation. This detailed breakdown demonstrates the model’s robust performance across various conditions and data subsets.

Table 17: Metrics for each dataset and data type for the Mistral-7B backbone.

Dataset	Data Type	AI F1	Human F1	Mean Accuracy	TPR@FPR=0.01
English-only	Article	0.9864	0.9829	0.9857	0.9798
	Factual text	0.9867	0.9803	0.9852	0.9796
	News	0.9881	0.9866	0.9877	0.9827
	Paper abstract	0.9934	0.9894	0.9923	0.9919
	Poetry	0.9886	0.9820	0.9863	0.9830
	Question	0.9840	0.9775	0.9833	0.9793
	Review	0.9778	0.9654	0.9767	0.9629
	Story	0.9926	0.9861	0.9910	0.9890
	Short-form text	0.9784	0.9439	0.9680	0.9434
Russian-only	Article	0.9883	0.9878	0.9881	0.9856
	Factual text	0.9836	0.9743	0.9810	0.9697
	News	0.9814	0.9805	0.9809	0.9739
	Poetry	0.9812	0.9825	0.9816	0.9746
	Question	0.9921	0.9760	0.9857	0.9849
	Review	0.9893	0.9841	0.9878	0.9853
	Story	0.9878	0.9846	0.9863	0.9841
	Short-form text	0.9784	0.8691	0.9551	0.8720
Bilingual	Article	0.9890	0.9878	0.9887	0.9854
	Factual text	0.9853	0.9778	0.9838	0.9772
	News	0.9844	0.9832	0.9839	0.9787
	Paper abstract	0.9927	0.9882	0.9905	0.9904
	Poetry	0.9853	0.9839	0.9850	0.9808
	Question	0.9888	0.9750	0.9837	0.9815
	Review	0.9839	0.9758	0.9826	0.9747
	Story	0.9894	0.9845	0.9878	0.9855
	Short-form text	0.9789	0.9206	0.9653	0.9056

Table 18: Metrics for each dataset and word bin for the Mistral-7B backbone.

Dataset	# Words	AI F1	Human F1	Mean Accuracy	TPR@FPR=0.01
English-only	<100	0.9747	0.9674	0.9725	0.9500
	100-400	0.9939	0.9891	0.9936	0.9914
	>400	0.9860	0.9813	0.9859	0.9778
Russian-only	<100	0.9746	0.9729	0.9740	0.9495
	100-400	0.9946	0.9860	0.9931	0.9941
	>400	0.9785	0.9932	0.9804	0.9706
Bilingual	<100	0.9747	0.9710	0.9734	0.9478
	100-400	0.9946	0.9881	0.9937	0.9935
	>400	0.9835	0.9882	0.9839	0.9751

Table 19: Metrics for each dataset and prompt type for the Mistral-7B backbone.

Dataset	Prompt Type	AI F1	AI Accuracy
English-only	Create	0.9674	0.9368
	Delete	0.9940	0.9880
	Expand	0.9986	0.9972
	Update	0.9966	0.9933
Russian-only	Create	0.9919	0.9839
	Delete	0.9820	0.9646
	Expand	0.9937	0.9874
	Update	0.9936	0.9875
Bilingual	Create	0.9844	0.9692
	Delete	0.9878	0.9759
	Expand	0.9968	0.9936
	Update	0.9959	0.9919

Table 20: Metrics for each dataset and generator (only top-5 generators with highest AI F1 metric and top-5 generators with lowest metric are reported) for the Mistral-7B backbone.

Dataset	Generator	AI F1	AI Accuracy
English-only (highest metrics)	llama-3.1-nemotron-70b-instruct-hf	0.9974	0.9949
	qwen/qwen3-32b	0.9965	0.9930
	databricks/dbrx-instruct	0.9949	0.9898
	cohereforai/c4ai-command-r-08-2024	0.9943	0.9887
	gpt-4.1-2025-04-14	0.9933	0.9868
English-only (lowest metrics)	qwen/qwen2.5-72b-instruct	0.9891	0.9784
	deepseek-ai/deepseek-r1-distill-qwen-32b	0.9873	0.9749
	mistralai/ministral-8b-instruct-2410	0.9777	0.9565
	zai-org/glm-4-32b-0414	0.9766	0.9543
	mistralai/magistral-small-2507	0.9630	0.9565
Russian-only (highest metrics)	gemma-1.1-7b-it	1.0000	1.0000
	qwen2-7b-instruct	0.9986	0.9972
	google/gemma-2-27b-it	0.9980	0.9960
	cohereforai/c4ai-command-r-08-2024	0.9979	0.9959
	phi-3-mini-128k-instruct	0.9973	0.9946
Russian-only (lowest metrics)	rugpt	0.9725	0.9465
	llama-7b	0.9717	0.9449
	o1-preview-2024-09-12	0.9620	0.9268
	jamba-v0.1	0.9322	0.8730
	microsoft/phi-3-medium-128k-instruct	0.9091	0.8333
Bilingual (highest metrics)	gemma-1.1-7b-it	1.0000	1.0000
	o1-mini-2024-09-12	1.0000	1.0000
	phi-3-mini-128k-instruct	0.9987	0.9973
	qwen/qwq-32b	0.9964	0.9982
	nvidia/llama-3.1-nemotron-70b-instruct-hf	0.9974	0.9949
Bilingual (lowest metrics)	o1-preview-2024-09-12	0.9268	0.9268
	llama-7b	0.9614	0.9256
	mistralai/magistral-small-2507	0.9608	0.9246
	microsoft/phi-3-medium-128k-instruct	0.9412	0.8889
	jamba-v0.1	0.9138	0.8413

K Creation of manually modified AI texts

The following protocol was applied consistently to both English and Russian sub-corpora to ensure cross-linguistic stylistic robustness. Initial AI-generated samples were extracted from the test set using a stratified sampling approach based on domain and document length. While we aimed for a balanced distribution, the final composition reflects the inherent constraints of the source data; for instance, certain domains (e.g., social media posts/tweets) naturally contain fewer long-form entries compared to news or academic documents.

Editors were tasked with manually intervening in the AI-generated text to disrupt predictable statistical patterns. To maximize the structural diversity of the resulting "hybrid" texts, editors followed three core principles:

- Length variation: editors maintained a rotating cycle of short, medium, and long-form texts to ensure the model remains length-agnostic.
- Variable integration density: the ratio of human-to-AI content was intentionally varied. In some instances, the AI was used only for a single concluding or introductory segment. In others, a "checkered" pattern was applied, where every second sentence was written or edited by a human.
- Structural randomization: intervention points were not restricted to sentence boundaries. Editors, for example, were encouraged to interweave human-written phrases (minimum five words) within AI-generated paragraphs.

Importantly, human editors did not produce explicit boundary annotations or define AI-generated spans as part of the annotation process. During editing, editors used a technical separator token (e.g., `_SEP_`) solely as a working delimiter to distinguish newly written or modified segments from the surrounding AI-generated text. These separators were not intended as boundary annotations and were not used directly in the experiments. The automatic construction of boundaries relied on a precise character-level comparison between the final edited version and the original AI-generated source text. Any character sequence preserved from the source text and located outside of the editor’s delimiters was definitively classified as AI-generated, and its exact offsets in the final text were recorded. Therefore, the final boundary-level segmentation used in downstream experiments was constructed automatically from the resulting edited texts and the known generation history, without manual specification, correction, or post-hoc adjustment.

By avoiding fixed patterns of intervention (e.g., only editing the beginning or only replacing full sentences), we created a sub-corpus that forces the detection models to focus on granular semantic and stylistic inconsistencies rather than simple structural markers.

L Annotation Examples

Figure 15 illustrates the two annotation schemas in LLMTrace: a binary classification sample (top) and a detection sample with character-level AI-interval annotations (bottom).

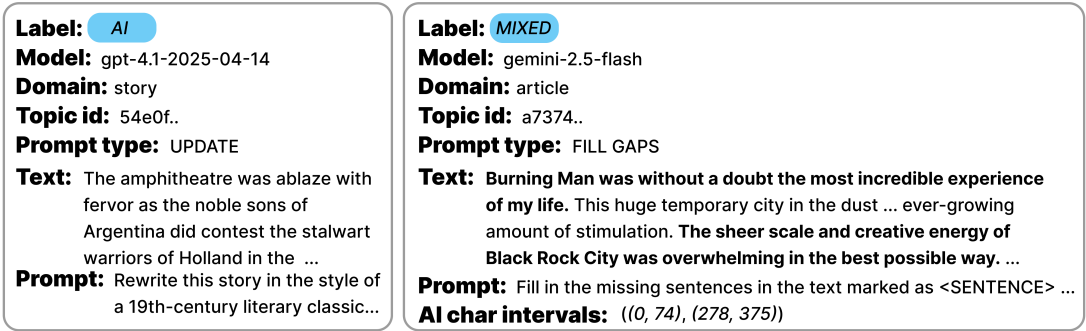


Figure 15: Example annotations from the LLMTrace English dataset: a classification sample (top) and a detection sample (bottom).

M Qualitative Examples

To illustrate the capabilities of our detection baseline, we present qualitative examples from the test set in Table 21. These samples demonstrate the performance of the GigaCheck (DN-DAB-DETR) model trained with a frozen Mistral-7B backbone across different mixed-authorship scenarios. We compare the Ground Truth (GT) annotations with the model’s predicted intervals to highlight both successful detections and boundary inconsistencies.

N Zero-Shot Detector Results

Table 22 reports binary classification performance for three zero-shot detectors evaluated on the same held-out test sets used for supervised baselines (Section 5.1). Binoculars and Fast-DetectGPT are evaluated with their default English-tuned thresholds. For DetectGPT we report only AUROC, as the method does not produce calibrated class scores suitable for direct accuracy/F1 computation.

The results confirm three conclusions. First, zero-shot performance is consistently poor, with even the best detector (Fast-DetectGPT) falling substantially short of the fine-tuned model. Second, Russian is markedly harder for all zero-shot methods: Binoculars in particular struggles severely, suggesting its perplexity-ratio signal is not calibrated for Russian vocabulary. Third, the difficulty is not an artifact of our evaluation protocol — DetectGPT’s AUROC remains close to chance level, confirming that meaningful detection on this dataset requires in-domain supervision.

Table 21: Comparison of **Ground Truth (GT)** labels and model **Outputs** for AI interval detection. **Bold text** marks AI-generated segments. While the model successfully detects spans of varying lengths, minor character-level boundary errors (e.g., splitting words) are observed.

Case 1: Manual Editing of AI Texts	Generator: OpenAI GPT-3.5
GT: In recent years, deep learning methods have achieved remarkable results in various computer vision tasks such as image classification, object detection, and segmentation. Non-local video denoising is a challenging problem that aims to reduce noise in video sequences while preserving the details and temporal consistency of the frames. In this paper, we present a novel approach for non-local video denoising using Convolutional Neural Networks (CNNs). Our method leverages the spatial and temporal information of video frames to perform non-local filtering. We trained a CNN to learn the mapping between noisy and clean video frames, which can effectively capture the underlying patterns in the video sequences. Results of the experiments show how out affroach can be more effective than contemporary methods in as subjective, so objective terms, also this method can evaluate quality in fields such as surveillance, medical imaging, and entertainment. Output: In recent years, deep learning methods have achieved remarkable results in various computer vision tasks such as image classification, object detection, and segmentation. Non-local video denoising is a challenging problem that aims to reduce noise in video sequences while preserving the details and temporal consistency of the frames. In this paper, we present a novel approach for non-local video denoising using Convolutional Neural Networks (CNNs). Our method leverages the spatial and temporal information of video frames to perform non-local filtering. We trained a CNN to learn the mapping between noisy and clean video frames, which can effectively capture the underlying patterns in the video sequences. Results of the experiments show how out affroach can be more effective than contemporary methods in as subjective, so objective terms, also this method can evaluate quality in fields such as surveillance, medical imaging, and entertainment.	
Case 2: Human Text Gap-Filling	Generator: Gemini-2.5 Flash
GT: It was my turn. It was finally time for the ritual to begin. I sat in a small abandoned church, on the south side of the town we had created. It was tall, honoring our God so very well. The stained-glass windows, though dusty and cracked, still glowed with a faint, ethereal light. I had a very simple white dress on, and I patted it down onto my pale legs. It was not like ones that I had seen on other Daughters in the villages around our town. I was excited, but thankfully not excited enough to break the ritual. My hands were clasped in my lap, trying to keep steady. And there I sat, and waited for the Father. He quickly came, and off we went, down the hallway into the back of the church. He led me slowly, carrying his candle. A candle, instead of a torch or a lantern. Output: It was my turn. It was finally time for the ritual to begin. I sat in a small abandoned church, on the south side of the town we had created. It was tall, honoring our God so very well. The stained-glass windows, though dusty and cracked, still glowed with a faint, ethereal light. I had a very simple white dress on, and I patted it down onto my pale legs. It was not like ones that I had seen on other Daughters in the villages around our town. I was excited, but thankfully not excited enough to break the ritual. My hands were clasped in my lap, trying to keep steady. And there I sat, and waited for the Father. He quickly came, and off we went, down the hallway into the back of the church. He led me slowly, carrying his candle. A candle, instead of a torch or a lantern.	

O Detection Error Analysis

This appendix expands the error analysis summarised in Section 6. The analysis is performed on the *mixed* subset of the LLMTrace test split (4,716 texts containing 13,589 ground-truth AI character intervals),

Table 22: Zero-shot detector performance on the binary classification test sets (Overall / English / Russian). DetectGPT reports AUROC only; dashes indicate unavailable metrics. Fine-tuned BIL Mistral-7B results are reproduced from Table 5 for reference.

Detector	Split	Accuracy	AI F1	Human F1	AUROC
Binoculars	Overall	0.493	0.282	0.610	–
	English	0.603	0.533	0.650	–
	Russian	0.412	0.022	0.580	–
DetectGPT	Overall	–	–	–	0.582
	English	–	–	–	0.613
	Russian	–	–	–	0.588
Fast-DetectGPT	Overall	0.663	0.692	0.626	–
	English	0.689	0.693	0.685	–
	Russian	0.643	0.692	0.576	–
<i>BIL Mistral-7B (fine-tuned)</i>	<i>Overall</i>	<i>0.985</i>	<i>0.986</i>	<i>0.980</i>	–
	<i>English</i>	<i>0.985</i>	<i>0.986</i>	<i>0.979</i>	–
	<i>Russian</i>	<i>0.984</i>	<i>0.986</i>	<i>0.980</i>	–

using the two bilingual interval-detection backbones from our main experiments (Section 5.2): **BIL Mistral-7B-DETR** and **BIL mmBERT-DETR**. The pipeline runs each detector once, caches per-sample predictions, and then aggregates them along several orthogonal axes (generator metadata and text-level properties) so that individual cuts can be re-derived without re-running inference.

O.1 Error Taxonomy

For one text, let $G = \{g_1, \dots, g_n\}$ be the GT character intervals and $P = \{p_1, \dots, p_m\}$ the predicted intervals. We construct the bipartite overlap graph $E = \{(i, j) : g_i \cap p_j \neq \emptyset\}$ and assign one of six labels to each interval based on its degree:

- **False negative.** A GT interval g_i with no overlapping prediction ($\deg_E(i) = 0$).
- **False positive.** A predicted interval p_j with no overlapping GT ($\deg_E(j) = 0$).
- **Good match.** GT g_i and prediction p_j form a 1-to-1 match ($\deg_E(i) = \deg_E(j) = 1$) with $\text{IoU}(g_i, p_j) \geq 0.8$.
- **Poor match.** As above but $\text{IoU}(g_i, p_j) < 0.8$ — the detector found the right region but the boundary is far off.
- **Over-segmentation.** A GT interval matched by multiple predictions ($\deg_E(i) > 1$); the detector fragmented one continuous AI span.
- **Under-segmentation.** A predicted interval matched by multiple GTs ($\deg_E(j) > 1$); the detector merged two AI spans separated by a short human insertion.

For every good match we additionally record the boundary offsets in characters, $(\text{start}_p - \text{start}_g, \text{end}_p - \text{end}_g)$, used in our boundary-bleeding analysis. From the per-text counts we derive interval-level metrics:

$$\text{precision} = \frac{N_{\text{good}} + N_{\text{poor}}}{N_{\text{good}} + N_{\text{poor}} + N_{\text{FP}}}, \quad \text{recall} = \frac{N_{\text{good}} + N_{\text{poor}}}{N_{\text{good}} + N_{\text{poor}} + N_{\text{FN}}},$$

$F_1 = 2PR/(P + R)$, and the *clean-match rate* $= N_{\text{good}}/|G|$ which ignores poor and segmentation-level matches.

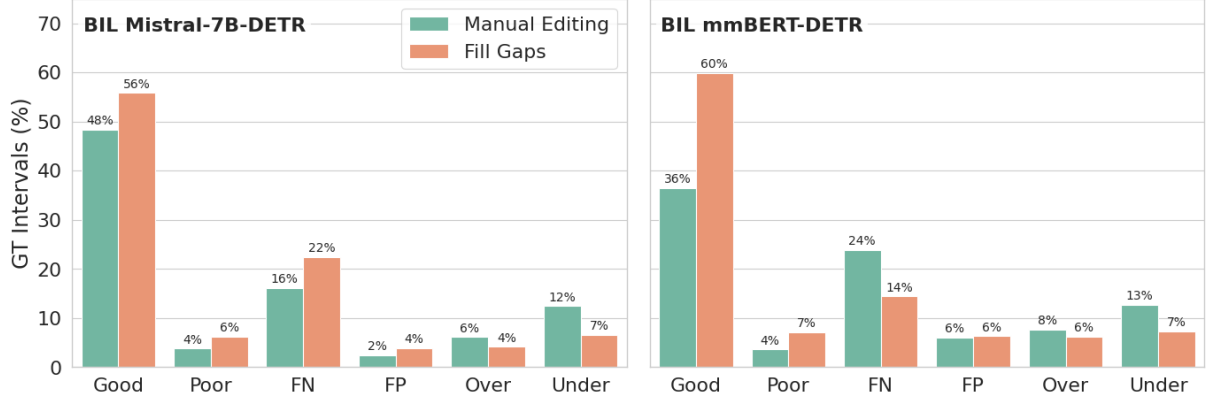


Figure 16: Aggregate distribution of the six error categories on the mixed subset, split by generation regime (manual editing vs. fill gaps) and by detector. Manual editing produces noticeably fewer good matches and more under-segmentation, confirming that targeted human edits disrupt detectable boundaries.

O.2 Generator Family Effects

Figure 5 (main body) reports the per-family error-type distribution for both detectors. Three observations are worth highlighting.

First, the frontier closed-source families (OpenAI, Gemini) together account for $\geq 90\%$ of the mixed test set; they reach interval-level F1 in the range $[0.82, 0.87]$ for both detectors and constitute the most reliable working point.

Second, legacy and small open-source families (ruGPT, early Llama) are catastrophically harder, with $F1 \leq 0.50$. The size-conditioned analysis (Section O.3) shows that this is a parameter-count effect rather than a family-specific one.

Third, several families show **detector-specific** behaviour: GigaChat reaches $F1 = 0.89$ on BIL Mistral-7B-DETR but only 0.47 on BIL mmBERT-DETR; Cohere drops from 0.64 to 0.41; Mistral drops from 0.77 to 0.52. This indicates that the two architectures pick up substantially different signals on these generators and motivates the cross-detector consistency analysis in Section O.7.

O.3 Generator Size

We bucket each generator by parameter count: *small* ($\leq 8B$), *medium* (8–30B), *large* ($> 30B$ open-weights), and *frontier-closed* (proprietary frontier models for which exact parameter counts are not publicly known). Per-bucket interval-level F1 is reported in Table 23.

size bin	n_{texts}	F1 (Mistral)	F1 (mmBERT)
small ($\leq 8B$)	32	0.475	0.453
medium (8–30B)	32	0.903	0.746
large ($> 30B$ open)	113	0.859	0.695
frontier-closed	4539	0.828	0.855

Table 23: Interval-level F1 by generator size bin. The small bucket is dominated by llama-7b, Ministral-8B, and ruGPT, all of which were rare in the detector training data.

The small bucket is the only one where both detectors agree on a sharp drop (F1 below 0.50). We caution that the small/medium bins are observational (32 mixed texts each), so absolute numbers should be read as indicative rather than conclusive; we report these bins for completeness, but the frontier-closed bin ($n = 4,539$) provides the only statistically powered comparison. Figure 18 shows the per-error-category breakdown for both detectors.

O.4 Generator Recency

Figure 19 shows per-generator F1 against release date together with OLS fits. For BIL Mistral-7B-DETR the slope is $+0.109$ F1/year ($R^2 = 0.40$, $p = 2.4 \times 10^{-4}$, $n = 29$); for BIL mmBERT-DETR it is $+0.045$

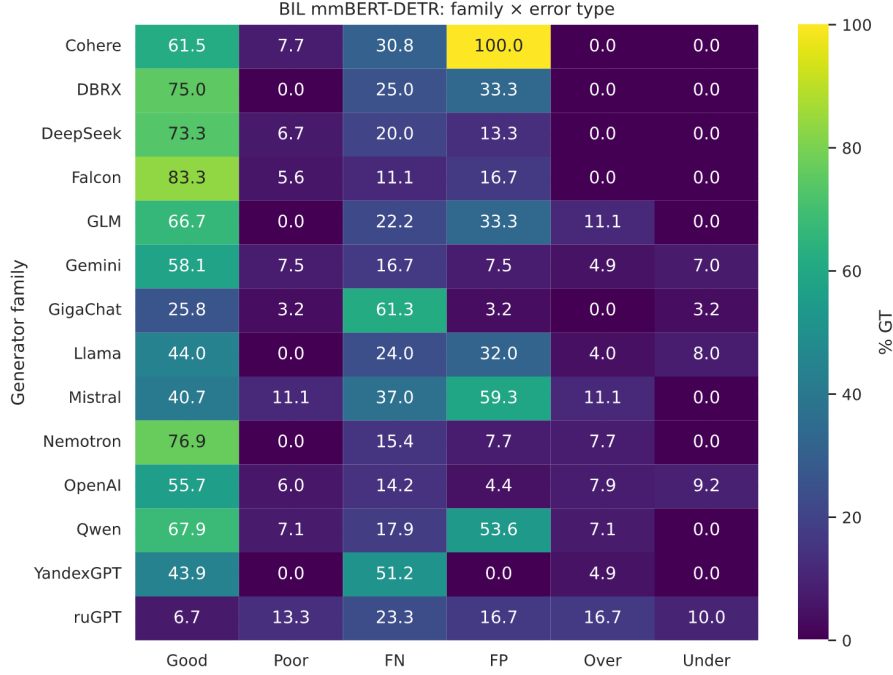


Figure 17: Per-family error-type distribution for the **BIL mmBERT-DETR** detector (companion to Figure 5). The frontier closed-source families dominate, but the mmBERT backbone shows substantially different behaviour on GigaChat, Cohere, and Mistral compared to the Mistral-7B backbone.

F1/year ($R^2 = 0.05$, $p = 0.25$, $n = 28$, not significant). The positive direction is the opposite of what one would expect if newer generators were inherently harder to detect.

The most plausible explanation is a **confound between recency and detector training distribution**. The older generators present in our test set — llama-7b (2023-02), ruGPT (2022-05) — are precisely the legacy small open-source models that were under-represented in the detector training corpus, while modern closed-source generators (gpt-4o, gemini-2.5-flash) match the training distribution well. Once the size-based out-of-distribution effect (Section 0.3) is taken into account, recency itself does not appear to drive detection difficulty in the date range covered by our dataset. We therefore advise against interpreting Figure 19 as evidence of a recency benefit.

A cleaner test of pure recency would require holding generator size and openness constant across release dates, which our current test set does not allow.

O.5 Generator Quality (Arena Elo Proxy)

As a public proxy for generator output quality we use Chatbot Arena Elo scores (snapshot late 2025). Elo is a noisy community-voting signal and our use of it is purely ordinal. For the 23 generators with available Elo we observe a positive slope: +0.08 F1 per +100 Elo for BIL Mistral-7B-DETR ($R^2 = 0.17$, $p = 0.052$) and +0.10 for BIL mmBERT-DETR ($R^2 = 0.32$, $p = 4.9 \times 10^{-3}$).

Higher-quality generators are detected slightly *better*, consistent with the hypothesis that more polished AI text exhibits a more distinctive stylistic signature. However, Elo and detector training distribution are themselves correlated — the highest-Elo generators (GPT-4o, Gemini, o1) are also the best represented in training data — so this is not a clean causal claim. We include this analysis to examine whether generator quality, as measured by a public proxy, correlates with detection difficulty.

O.6 Text-level Factors

Text length. Figure 21 reports precision, recall, and clean-match rate as a function of text length. Precision stays high (≥ 0.88) across all length bins for both detectors, while recall collapses on long texts: from 0.84 (≤ 200 words) to 0.15 (≥ 2000 words) for BIL Mistral-7B-DETR. The detector silently misses AI spans in long contexts rather than producing spurious predictions. Per-sample Spearman

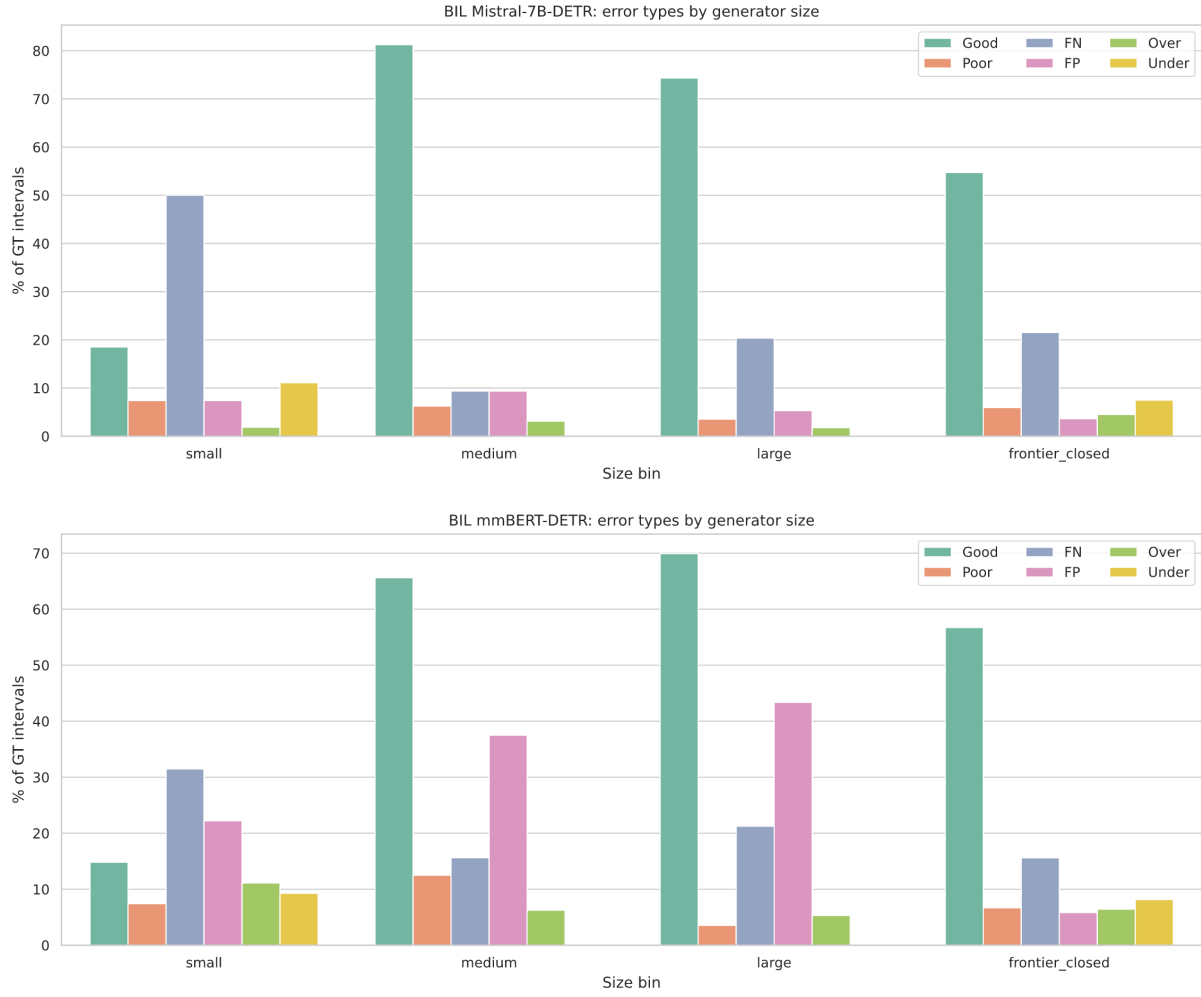


Figure 18: Per-error-category share by generator size bin, BIL Mistral-7B-DETR (top) and BIL mmBERT-DETR (bottom).

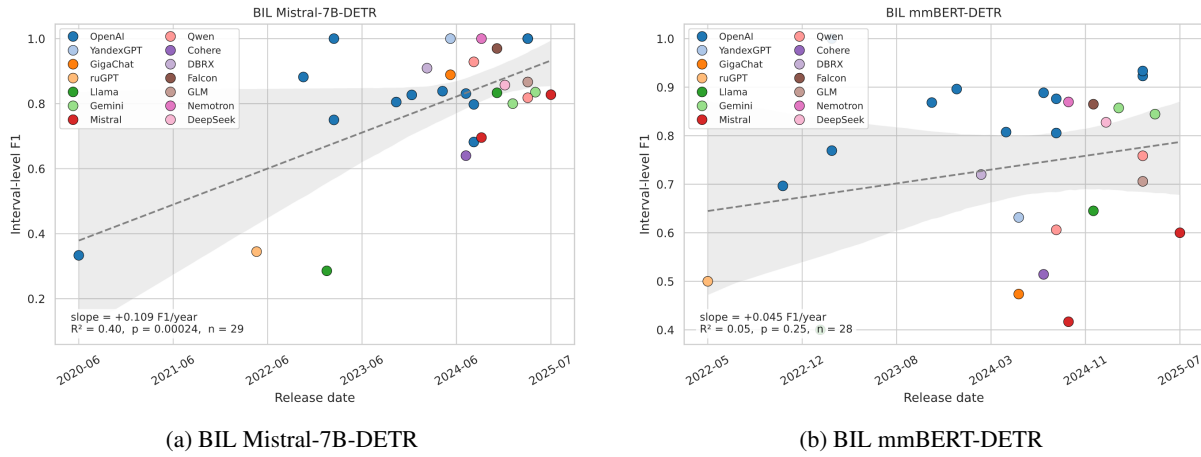


Figure 19: Per-generator interval-level F1 vs. release date with OLS fits and 95% CI bands. (a) Mistral-7B backbone: positive slope (+0.11 F1/year, $p = 2.4 \times 10^{-4}$) driven by legacy small open-source generators (llama-7b, ruGPT) being out-of-distribution for the detector. (b) mmBERT backbone: same direction but the slope is not significant ($p = 0.25$). Both panels are explained by the recency–training-distribution confound discussed above.

correlation between text length (words) and clean-match rate is $\rho = -0.31$ ($p < 10^{-100}$, $n = 4716$) for BIL Mistral-7B-DETR and $\rho = -0.27$ for BIL mmBERT-DETR.

F1 vs Arena Elo (proxy; snapshot $\approx$ 2025-Q4)

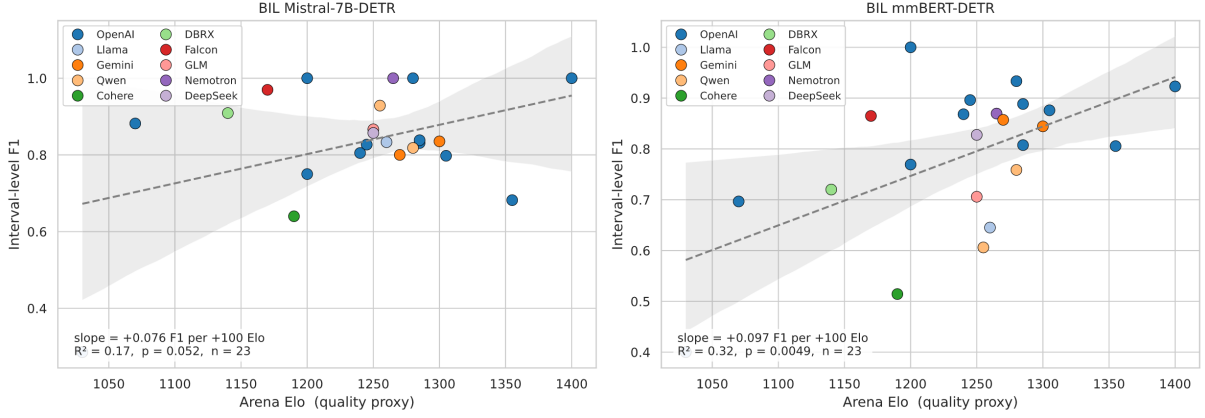


Figure 20: Per-generator F1 vs. Chatbot Arena Elo (proxy for generation quality), with OLS fit and 95% CI band.

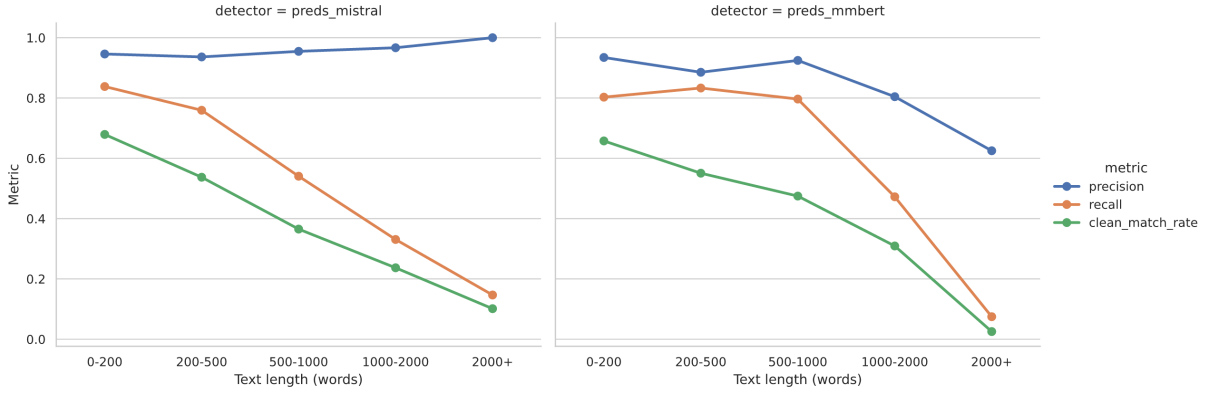


Figure 21: Precision, recall, and clean-match rate vs. text length (words), for both detectors.

Number of GT intervals per text. F1 stays in $[0.85, 0.89]$ for texts with 1–5 GT intervals and drops to ~ 0.75 for texts with 6+ intervals (Figure 22). The dominant degradation mode at high interval count is under-segmentation: the rate of GT intervals being merged into one prediction grows from 0% (single-interval texts) to roughly 10% (6+ intervals). Per-sample Spearman ρ between interval count and clean-match rate is -0.38 for BIL Mistral-7B-DETR ($p < 10^{-150}$).

Domain. F1 varies only mildly across the nine data_type categories (Figure 23). Both detectors find templated short forms (paper_abstract, short_form, review) easier and creative long forms (story, article) harder. Poetry shows a notable detector-architecture flip: it is the worst domain for BIL Mistral-7B-DETR ($F1 = 0.80$) but among the best for BIL mmBERT-DETR ($F1 = 0.87$), suggesting that the two architectures rely on different stylistic cues for highly structured text.

Language. The bilingual split shows no systematic disadvantage for either language (Figure 24); BIL Mistral-7B-DETR is slightly better on English ($\Delta F1 = +0.02$) while BIL mmBERT-DETR is slightly better on Russian ($\Delta F1 = +0.03$). All differences fall within ± 3 F1 points and do not change the qualitative conclusions of the previous subsections.

O.7 Cross-detector Agreement

To separate generator-intrinsic difficulty from architecture-specific failure modes, we compute Spearman and Pearson correlations between the two detectors’ per-group F1 across multiple groupings (Table 24). The per-model Spearman of 0.40 ($p = 0.04$) is statistically significant but modest in magnitude — the two architectures do not fully agree on which generators are hard.

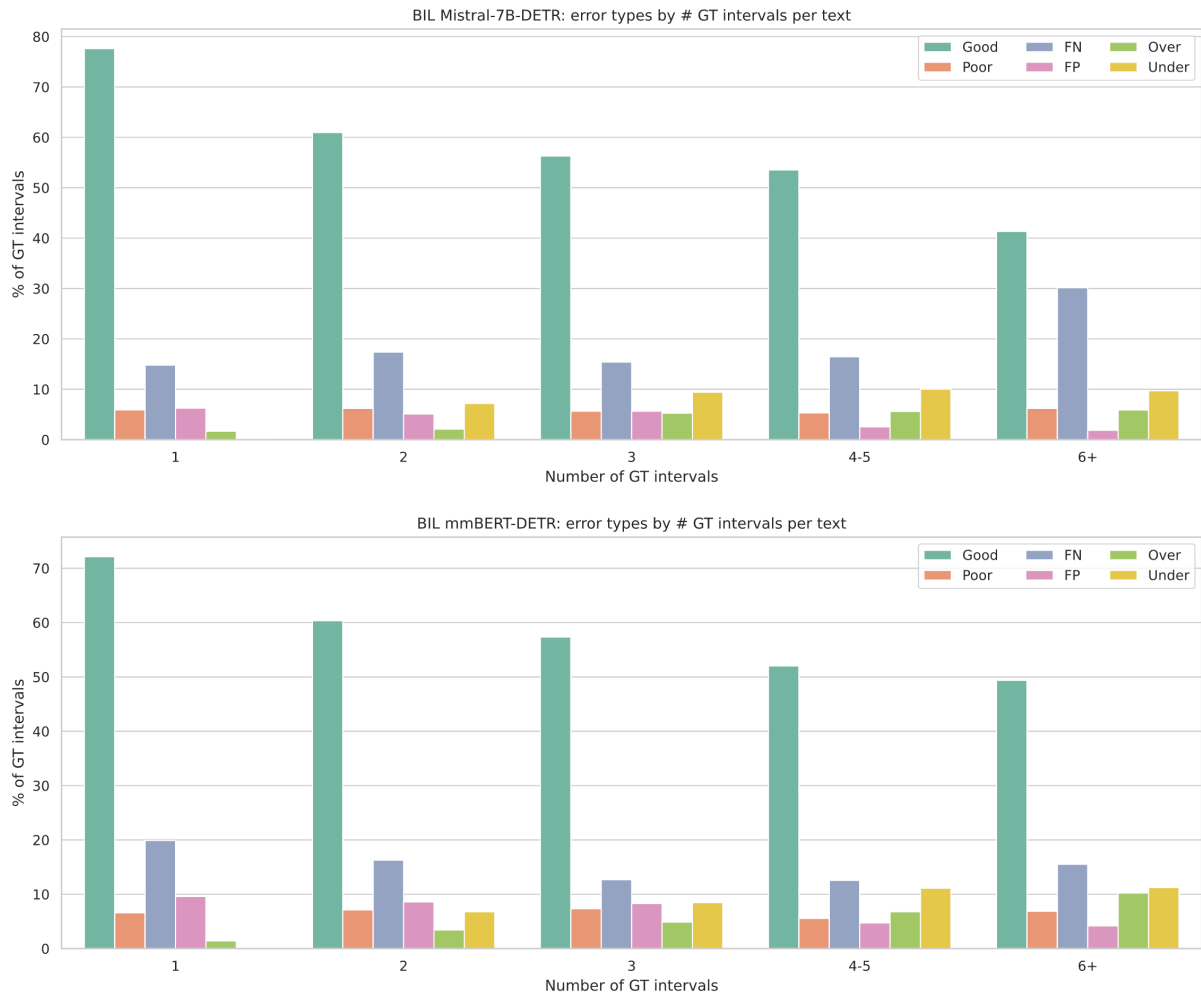


Figure 22: Per-error-category share by number of GT intervals per text, BIL Mistral-7B-DETR (top) and BIL mmBERT-DETR (bottom).

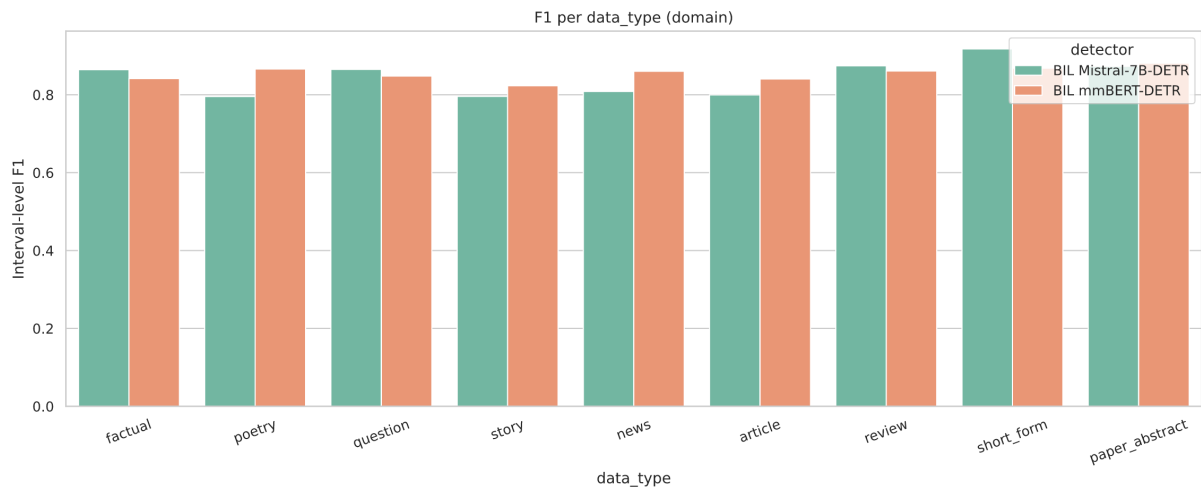


Figure 23: Interval-level F1 per data_type (domain proxy), both detectors.

We interpret this as evidence that detection difficulty has both **data-intrinsic (generator-driven)** and **architecture-specific** components. Practically: detector behaviour on a previously unseen generator family cannot be reliably extrapolated from a single architecture’s per-family profile. Reporting analyses for at least two detector backbones, as we do throughout this appendix, is therefore important for any

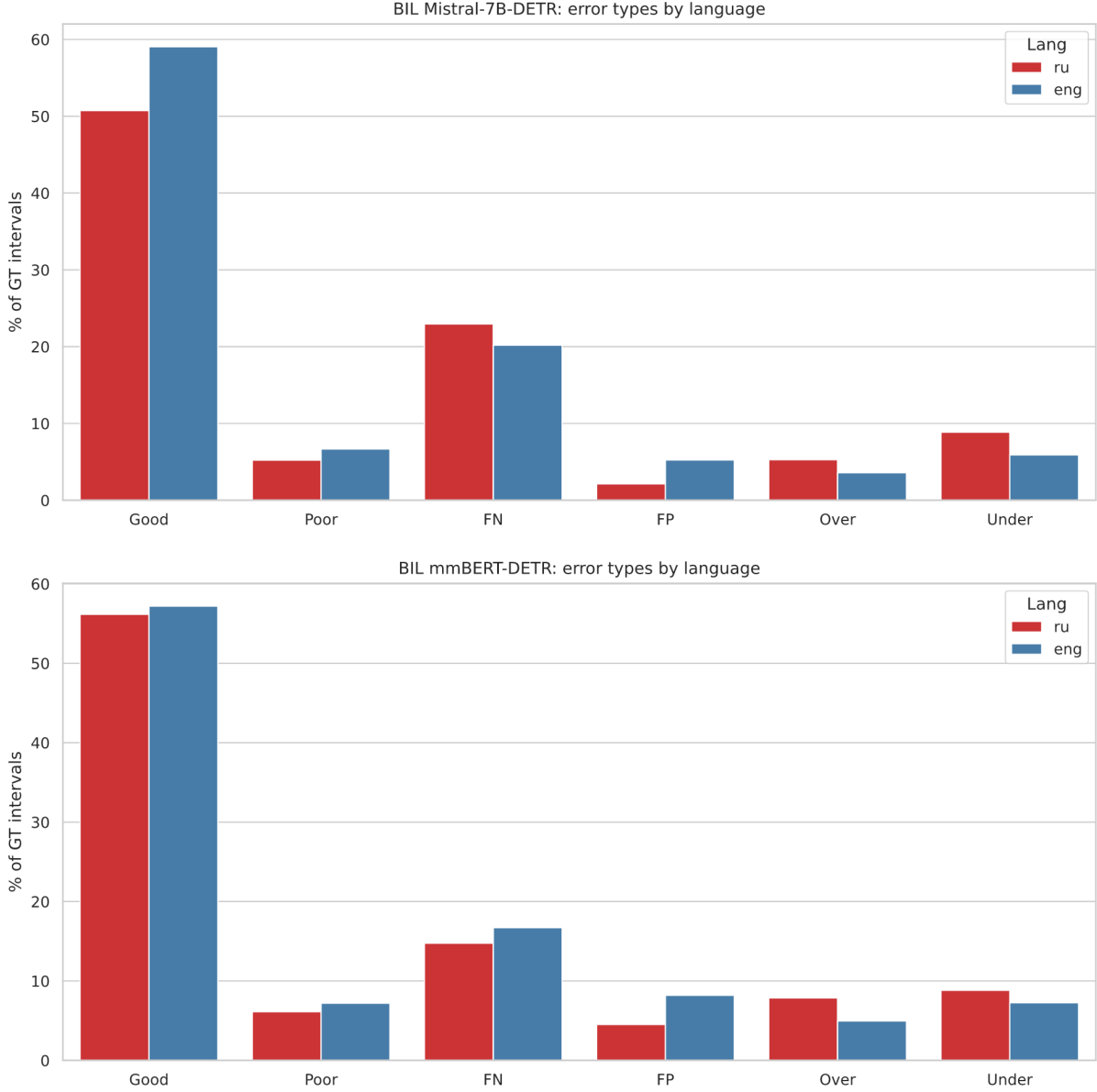


Figure 24: Per-language error-category breakdown for BIL Mistral-7B-DETR (top) and BIL mmBERT-DETR (bottom).

claim of the form “generator X is hard / easy”.

O.8 Transferability to External Benchmarks

To assess transferability beyond the LLMTrace distribution, we evaluate both detectors zero-shot (no target-domain training) on the RoFT (Dugan et al., 2023) and RoFT-ChatGPT (Kushnareva et al., 2023) boundary benchmarks. Following Tolstykh et al. (2026), we keep only the top-1 predicted interval per text and project its start to a sentence index using the same character-to-sentence mapping; we report SoftAcc1 and MSE.

Table 25 reports our results alongside the baselines from Kushnareva et al. (2023) and the GigaCheck-DETR† variant of Tolstykh et al. (2026), which shares our pre-trained Mistral-7B-v0.3 backbone but trains its DETR head directly on each target benchmark. Despite using no target-domain training, our Mistral detector matches the human baseline on RoFT (SoftAcc1 0.39 vs 0.40) and outperforms both topology-based baselines (PHD+TS, TLE+TS) on RoFT-ChatGPT — all of which were trained on the target. It expectedly trails the methods that fully fit the target benchmark (RoBERTa variants,

grouping	n	Spearman ρ	p	Pearson r
by model	28	+0.40	0.04	0.59
by family	14	+0.45	0.11	0.55
by size bin [†]	4	+0.40	0.60	0.87
by release y-m	18	-0.20	0.42	0.61
by Arena Elo bin [†]	5	0.00	1.00	0.42

Table 24: Cross-detector agreement (BIL Mistral-7B-DETR vs. BIL mmBERT-DETR) on per-group F1.

[†]Correlations for groupings with $n < 10$ are reported for completeness but lack statistical power.

GigaCheck-DETR[†]), and the larger gap on RoFT is consistent with the generator-coverage analysis above: legacy GPT-2/CTRL generators dominating RoFT are under-represented in LLMTrace, whereas the modern generators in RoFT-ChatGPT match our training distribution. The smaller mmBERT backbone transfers noticeably worse despite matching Mistral within $\sim 1\%$ mAP in-distribution (Table 3), indicating that backbone capacity becomes more critical out-of-distribution. Overall, LLMTrace-trained detectors transfer non-trivially to external benchmarks without any target-domain fine-tuning, and the residual gap to fully supervised models tracks rather than contradicts the in-distribution diagnosis — pointing to broader generator coverage, especially of legacy and small open-source models, as the main lever for closing it.

Table 25: Zero-shot transfer to RoFT and RoFT-ChatGPT. All baselines except ours are trained or calibrated on the target benchmark; their numbers are taken from Kushnareva et al. (2023), except GigaCheck-DETR[†] which is taken from Tolstykh et al. (2026) and shares our pre-trained Mistral-7B-v0.3 backbone (its DETR head is trained on each target benchmark). SoftAcc1 and MSE are computed on the top-1 predicted interval, projected to sentence indices following Tolstykh et al. (2026).

Method	RoFT		RoFT-ChatGPT	
	SoftAcc1	MSE	SoftAcc1	MSE
RoBERTa + SEP	0.80	2.63	0.79	3.06
Phi-1.5 Perpl. + GB	0.45	6.11	0.71	3.07
Phi-1.5 Perpl. + LR	0.50	11.9	0.73	4.77
PHD + TS ML	0.46	14.40	0.36	14.45
TLE + TS Binary	0.30	22.23	0.35	18.52
Human baseline	0.40	13.88	—	—
GigaCheck-DETR [†]	0.81	2.77	0.80	1.93
BIL Mistral-DETR (zero-shot)	0.39	15.58	0.58	7.15
BIL mmBERT-DETR (zero-shot)	0.32	21.67	0.48	13.04

Beyond boundary detection, we additionally evaluate the binary classifiers from Table 5 on the MAGE benchmark (Li et al., 2023). Following Tolstykh et al. (2026), we report AvgRec (the per-class accuracy average) and AUROC on the *Arbitrary-domains & Arbitrary-models* test split, which is the standard in-distribution test set used by MAGE-trained baselines. Without any MAGE training, our BIL Mistral-7B classifier retains $\approx 71\%$ of the in-distribution trained classifier’s AvgRec (0.68 vs. 0.96 (Tolstykh et al., 2026)) and $\approx 80\%$ of its AUROC (0.79 vs. 0.99). The BIL mmBERT classifier drops further (AvgRec 0.59, AUROC 0.63), reproducing the smaller-backbone deficit observed above on RoFT (Table 25); the two transferability experiments thus agree that backbone capacity, not architectural family, is the dominant lever once the target distribution drifts away from training. Full numbers and the trained baselines are listed in Table 26.

Table 26: Zero-shot transfer of the binary classifiers from Table 5 to MAGE (Li et al., 2023) on the *Arbitrary-domains & Arbitrary-models* test split (n=56,819). Trained baseline numbers are taken from Tolstykh et al. (2026); the Longformer row in turn cites the value reported there from Li et al. (2023).

Method	AvgRec	AUROC
Longformer (Beltagy et al., 2020) (trained on MAGE)	0.91	0.99
GigaCheck Mistral-7B (Tolstykh et al., 2026) (trained on MAGE)	0.96	0.99
BIL Mistral-7B classifier (zero-shot)	0.68	0.79
BIL mmBERT classifier (zero-shot)	0.59	0.63