C-HAT-BENCH: BENCHMARKING CHINESE AI-TEXT DETECTION BEYOND FULLY GENERATED TEXT

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) increasingly participate in writing by modifying or extending human drafts, causing machine involvement to vary in both form and extent. Yet most Machine-Generated Text (MGT) detectors are evaluated only on fully human-written versus fully AI-generated text. Because human–AI collaboration can weaken or redistribute cues associated with machine generation, strong performance under this binary setting may overstate detector reliability. This mismatch remains underexplored in Chinese: detection cues are shaped by tokenization and language-specific text distributions, yet controlled resources spanning production settings, domains, and generators remain limited. To fill this gap, we present a **C**hinese **H**uman-**AI** Collaborative **T**ext Detection **Bench**mark (**C-HAT-Bench**), a unified benchmark that links $5,000$ human-written source texts from five domains to more than $240,000$ variants produced using six generative models under *Prefix-Conditioned Continuation* as a reference setting and three collaborative production modes. We evaluate 21 detectors through four protocols spanning zero-shot and pretrained supervised document-level detection, boundary localization, and cross-condition generalization. Relative to Prefix-Conditioned Continuation, mean AUROC across document-level detectors is 12.0% lower on the collaborative production modes, with the largest detector-specific relative decrease reaching 44.4%. Transfer across collaborative production modes is also asymmetric, indicating that performance in a given production setting is not a reliable predictor of performance in other production settings.[1]

1 INTRODUCTION

Recent advances in Large Language Models (LLMs) (Pan et al., 2024; Zhao et al., 2026) have substantially enhanced text-generation capabilities, contributing to their broader adoption in writing and information production. As LLMs are increasingly integrated into writing workflows, the resulting texts may reflect contributions from both human authors and generative models, making the extent of machine involvement difficult to determine. Reliable machine-generated text (MGT) detection is therefore needed to support provenance assessment, particularly when generated content is used to disseminate misinformation or facilitate academic misconduct (Brown et al., 2020; Wang et al., 2024a; Wu et al., 2025; Quaremba et al., 2026).

Yet reliable MGT detection remains challenging because the process through which a text is produced is not directly observable, while the cues associated with machine generation depend on the generator and its role in writing. Prevailing benchmarks still emphasize binary discrimination between fully human-written and fully AI-generated text, typically using outputs elicited by generic prompts (Wu et al., 2025; Wang et al., 2024a; Quaremba et al., 2026). Although this setting provides a useful controlled baseline, it does not capture human-AI collaborative writing, in which machine involvement varies in form and extent. Consequently, performance measured on fully AI-generated text does not, by itself, establish detector reliability under human-AI collaboration (Wang et al., 2024a; Quaremba et al., 2026).

Moreover, process-aware MGT evaluation in Chinese must account for detection cues that are sensitive to tokenization and language-specific text distributions. Chinese lacks explicit word boundaries,

[1] Code and data are available at: https://anonymous.4open.science.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

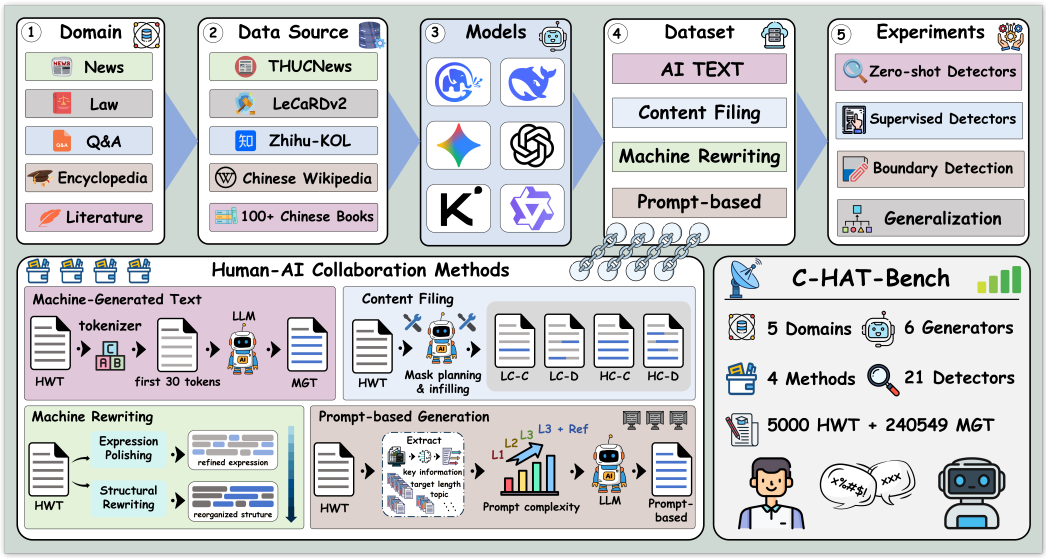

Figure 1: Overview of the C-HAT-Bench framework.

and its writing conventions vary substantially across domains (Tsang et al., 2025; Qing et al., 2026). Yet existing Chinese benchmarks do not enable controlled comparisons of human–AI collaboration modes across domains and generators (Guo et al., 2023; Macko et al., 2023; Wang et al., 2024b; Qing et al., 2026). A unified benchmark is therefore needed to assess detector reliability across human–AI writing processes in Chinese.

To enable such evaluation, we construct the Chinese Human–AI Collaborative Text Dataset (**C-HAT-240K**), which links each human-written source text to multiple machine-involved variants. C-HAT-240K comprises $5,000$ source texts from five domains—news, law, question answering, encyclopedic writing, and literature—and $240,539$ variants produced with six LLMs. As shown in Figure 1, the construction includes Prefix-Conditioned Continuation as a reference setting and three collaborative production modes: Content Filling, Machine Rewriting, and Prompt-based Generation. The reference setting retains a human-written prefix, whereas the collaborative modes vary generated-span coverage and placement, rewriting intensity, and prompt guidance. This source-linked design enables controlled comparisons of detector performance across production processes.

Building on C-HAT-240K, we introduce a **C**hinese **H**uman-**AI** Collaborative **T**ext Detection **Bench**mark (**C-HAT-Bench**), a unified framework for evaluating 21 detectors under four protocols. The document-level evaluation comprises separate protocols for 10 zero-shot detectors and 7 pretrained supervised detectors across all production settings. A third protocol assesses 2 detectors on the Content Filling subset for localizing boundaries between human-written and machine-generated content. The fourth examines cross-condition generalization by training 2 supervised detectors on one condition at a time and evaluating their transfer under changes in domain, production setting, or generator. The main contributions of this work are threefold:

- **Dataset.** We construct C-HAT-240K, a Chinese dataset that links 5,000 human-written source texts from five domains to more than 240,000 variants produced using six generative models. The variants cover one reference setting and three collaborative production modes;
- **Unified evaluation benchmark.** We introduce **C-HAT-Bench**, a unified framework for evaluating 21 detectors through four protocols. These comprise separate document-level protocols for ten zero-shot and seven pretrained supervised detectors, a boundary-localization protocol with two detectors, and a cross-condition generalization protocol with two supervised detectors;
- **Multi-dimensional analysis.** We analyze detector sensitivity to the proportion and placement of machine-generated content, the degree of model intervention, and the level of guidance provided by prompts. The analysis further examines boundary localization and transfer across production settings, domains, and generators.

Key findings. Relative to Prefix-Conditioned Continuation, mean AUROC across detectors is 12.0% lower on the three collaborative production modes, with the largest detector-specific relative decrease reaching 44.4%. Boundary matching improves under dispersed filling, while transfer across collaborative production modes is asymmetric, indicating that performance under one production setting does not reliably predict performance under another.

2 Related Work

AIGC Text Detection Methods AIGC text detection has progressed from document-level source classification to fine-grained attribution in human–AI collaborative text. Early work focused on statistical traces in generated text: Gehrmann et al. (2019) visualized token probabilities and ranks, while Verma et al. (2024) learned classifiers from probability-derived features. As reliance on labeled data became a central limitation, Mitchell et al. (2023) introduced probability-curvature-based detection, followed by log-rank, conditional-curvature, perplexity-based, and query-based methods (Su et al., 2023; Bao et al., 2024; Hans et al., 2024; Zhu et al., 2023). A complementary line embeds detectable signals during generation: Kirchenbauer et al. (2023) proposed a statistical watermark for language models, and Dathathri et al. (2024) extended watermarking to scalable identification of model outputs. Subsequent studies addressed robustness under domain shifts, adversarial manipulation, and machine rewriting (Bhattacharjee et al., 2023; Hu et al., 2023; Kuznetsov et al., 2024; Chen et al., 2025; Hao et al., 2025). With the emergence of mixed-origin text, Lee et al. (2022) introduced a human–AI collaborative writing dataset, while later studies investigated machine revision, sentence-level attribution, boundary segmentation, and generated-span localization (Zhang et al., 2024; Wang et al., 2023; Zeng et al., 2024; Jiang et al., 2025; Li et al., 2026; Tolstykh et al., 2026). These developments move detection beyond document-level authorship, but realistic collaboration processes remain insufficiently covered.

Chinese AIGC Corpora Chinese NLP research first established broad linguistic resources for different levels of analysis. Xue et al. (2005) introduced the Penn Chinese Treebank for syntactic annotation, while Zhou & Xue (2014) developed the Chinese Discourse TreeBank for discourse-relation analysis. Large-scale pretraining resources were later provided by Yuan et al. (2021). Task-oriented datasets further expanded domain coverage, including legal retrieval through LeCaRDv2 (Li et al., 2024a). For AIGC detection, Guo et al. (2023) constructed a Chinese question-answering corpus pairing human and ChatGPT responses, while MULTITuDE and M4 broadened the coverage of languages, domains, and generators (Macko et al., 2023; Wang et al., 2024b). C-ReD further introduced Chinese detection data derived from real-world-inspired prompts across domains and models (Qing et al., 2026). These resources provide important foundations, but they do not jointly control collaboration modes, machine-involvement levels, and source-linked local authorship.

MGT Detection Benchmarks MGT benchmarks have consequently moved from source identification toward evaluation under changing data conditions. Uchendu et al. (2021) established document-level detection and generator attribution, after which MAGE and MGTBench broadened comparisons across domains, generators, and detection methods (Li et al., 2024b; He et al., 2024). Robustness benchmarks such as BUST, RAID, and DetectRL then examined detector behavior under perturbations, adversarial attacks, evasion settings, and real-world generation conditions (Cornelius et al., 2024; Dugan et al., 2024; Wu et al., 2024). For mixed-origin text, M4GT-Bench modeled a human-written prefix followed by machine-generated continuation, whereas TSM-BENCH evaluated detection in practical Wikipedia editing tasks (Wang et al., 2024a; Quaremba et al., 2026). However, existing benchmarks still lack a unified Chinese setting that connects controlled collaboration construction with local boundary localization and cross-condition generalization.

3 C-HAT-Bench

3.1 Dataset Construction

Data Collection. C-HAT-240K covers five domains, with human-written source texts for each domain drawn from a distinct dataset. News texts are drawn from THUCNews[2]. The Law, Q&A,

[2]Dataset available at `http://thuctc.thunlp.org`.

and Encyclopedia texts come from LeCaRDv2 (Li et al., 2024a), Zhihu-KOL[3], and the Chinese Wikipedia corpus on ModelScope[4], respectively. Literature texts come from a locally curated collection of 102 modern Chinese literary works. Detailed data extraction procedures and dataset details are provided in Appendix A.1.

Data Processing. Before C-HAT-240K construction, we extract and clean the human-written texts. For domains other than Literature, we target 1,000 texts per domain by starting from a central text-length target measured in characters, progressively expanding the selection range toward shorter and longer texts, and relaxing the Chinese-character proportion threshold when necessary without falling below a preset minimum. The Literature domain is processed separately by segmenting works at paragraph and chapter boundaries, removing non-content material, and retaining 1,000 continuous passages. All selected texts are then normalized, stripped of metadata artifacts and malformed fragments, and deduplicated; detailed procedures are provided in Appendix A.2.

Dataset Building. We organize the machine-involved variants into four construction types, each derived from a human-written source and instantiated with the six generative models. Let x and y denote the source and constructed texts, respectively. $\mathcal{M}_\theta(u, p)$ denotes a generative model with parameters θ that produces text from input u under prompt p, and $\oplus$ denotes string concatenation. Detailed descriptions of the generative models are provided in Appendix A.3, while detailed construction rules and prompt templates are given in Appendix A.4.

- **Type I: Machine-Generated Text.** We retain the first 30 tokens of each source text under the Qwen2.5-3B-Instruct tokenizer and use them as the context for continuation:
$$y_\theta^{\mathrm{MGT}}(x) = \mathrm{Pref}_{30}^{\tau}(x) \oplus \mathcal{M}_\theta(\mathrm{Pref}_{30}^{\tau}(x), p_{\mathrm{cont}})\,. \tag{1}$$
Here, $\mathrm{Pref}_{30}^{\tau}(x)$ is the retained human-written prefix, p_{cont} requests a model continuation. In the following experiments, MGT refers to this prefix-conditioned reference condition.
- **Type II: Content Filling.** We replace selected source spans with generated text while preserving the remaining human-written content. The four conditions combine low or high coverage with contiguous or dispersed spans: *1. LC-C:* low coverage in one contiguous suffix; *2. LC-D:* low coverage in two or three dispersed spans; *3. HC-C:* high coverage in one contiguous suffix; and *4. HC-D:* high coverage in three or four dispersed spans. Low and high coverage correspond to 15–35% and 50–75% of the source characters, respectively:
$$y_{\theta,t}^{\mathrm{CF}}(x) = h_{t,0} \oplus \bigoplus_{j=1}^{K_t} [\mathcal{M}_\theta(\tilde{x}_{t,j}, p_{t,j}) \oplus h_{t,j}]\,, \quad t \in \mathcal{T}_{\mathrm{CF}}. \tag{2}$$
Here, $\mathcal{T}_{\mathrm{CF}}$ contains the four conditions, $h_{t,j}$ denotes an unchanged source segment, and $\tilde{x}_{t,j}$ is the masked source after prior replacements are inserted. The value of K_t and span layout are determined by condition t, while $p_{t,j}$ specifies the target span and its length requirement.
- **Type III: Machine Rewriting.** We provide the complete source text to the model under two rewriting settings:
$$y_{\theta,c}^{\mathrm{MR}}(x) = \mathcal{M}_\theta(x, p_c), \quad c \in \{\mathrm{LR}, \mathrm{DR}\}. \tag{3}$$
Here, *LR* (Light Rewriting) performs local wording and fluency adjustments while retaining the original organization; *DR* (Deep Rewriting) allows broader changes to expression and organization while preserving the source topic, main facts, and domain style.
- **Type IV: Prompt-based Generation.** GPT-5.5 first extracts a shared structured representation from each source, after which level-specific fields and writing instructions are assembled into the generation prompt:
$$y_{\theta,\ell}^{\mathrm{PG}}(x) = \mathcal{M}_\theta(F_\ell(E_{\text{GPT-5.5}}(x)) \oplus r_\ell, p_\ell)\,, \quad \ell \in \{\mathrm{L1}, \mathrm{L2}, \mathrm{L3}, \mathrm{L3+Ref}\}. \tag{4}$$
Here, $E_{\text{GPT-5.5}}$ extracts the shared structured fields, F_ℓ selects and formats the fields used at level ℓ, p_ℓ specifies the corresponding writing instructions, and r_ℓ denotes an optional human-written reference from the same domain. For the first three levels, r_ℓ is empty. *L1* uses task information only; *L2* adds the key content to the task information; *L3* further incorporates domain-specific constraints; and *L3+Ref* retains the L3 configuration while additionally providing a different human-written text from the same domain as r_ℓ.

[3]Dataset available at `https://huggingface.co/datasets/wanqiu6/Zhihu-KOL`.
[4]Dataset available at `https://www.modelscope.cn/datasets/caolong/zhwiki`.

Dataset Quality Improvement. After constructing the four machine-involved subsets, we first retain only samples whose generated-to-source character-length ratio falls within $[0.6, 1.4]$. We then remove empty or malformed outputs, API error responses, prompt leakage, residual Markdown/HTML wrappers, severe encoding corruption, and clearly truncated or degenerated generations. The resulting corpus contains 245,539 records, including 5,000 shared HWT samples and 240,539 machine-involved samples spanning the benchmark's four production settings, as summarized in Fig. 2. Finally, the retained records are mapped to a unified schema, with text fields and metadata such as source identifiers, domains, production types, subtype labels, and generators standardized and validated. Detailed dataset statistics are provided in Appendix A.5.

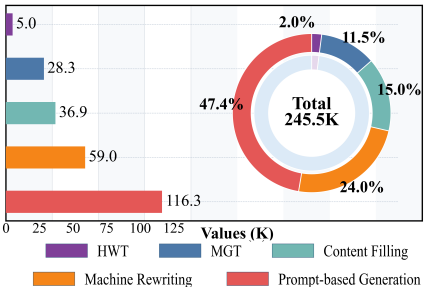

Figure 2: Composition of C-HAT-240K across production settings.

3.2 Benchmark Setting

Evaluation Tasks. To characterize detector behavior under different sources of variation, C-HAT-Bench evaluates four complementary tasks:

- **Detection performance across human–AI collaboration conditions.** We evaluate document-level detection across collaboration conditions to measure how the form and extent of machine involvement affect detectability.
- **Detection sensitivity across text domains.** We compare detectors within each text-production setting to determine how detection performance depends on the source domain.
- **Boundary localization in human–AI collaborative texts.** We identify transitions between human-written and machine-generated segments to assess whether detectors can recover fine-grained authorship changes.
- **Cross-condition generalization.** We evaluate transfer across domains, collaboration modes, and generation models to measure detector robustness under distribution shifts.

Detectors. To cover general and task-specific detection settings, we use 10 zero-shot and 7 off-the-shelf supervised detectors. The first two experiments evaluate the same detector sets. Boundary localization uses WCP and DeBERTa–BiGRU–CRF, while generalization experiments use AIGC-detector-zhv2 and IMBD. Detailed information on the detectors is provided in Appendix B.

Evaluation Metrics. To align each metric with the corresponding task, text-level detection is evaluated using AUROC (Fawcett, 2006). Accuracy, and F1-score (Sokolova & Lapalme, 2009) at a fixed 5% FPR operating point. Boundary localization uses sentence-level Cohen's κ (Cohen, 1960), WindowDiff (Pevzner & Hearst, 2002), and Boundary F1@1 (Zeng et al., 2024). For generalization, we report accuracy at cell-specific thresholds maximizing Youden's J on labeled test samples, a criterion also used for zero-shot calibration in TSM-Bench (Quaremba et al., 2026).

4 Experimental Results & Discussion

Our evaluation uses four questions: **RQ1 (Detection Performance across Human–AI Collaboration Conditions):** How does detector performance change with the form and degree of human–AI collaboration? (Sec. 4.1) **RQ2 (Domain sensitivity):** How do text domains affect the detection of human–AI collaborative text? (Sec. 4.2) **RQ3 (Boundary localization):** Can existing detectors accurately localize the boundaries between human-written and machine-generated content in collaborative text? (Sec. 4.3) **RQ4 (Cross-distribution generalization):** Can supervised detectors generalize across shifts in domain, collaboration type, and generation model? (Sec. 4.4) Detailed experimental settings are provided in Appendix C.

4.1 Detection Performance across Human–AI Collaboration Conditions (RQ1)

Experiment design. We evaluate 10 zero-shot and 7 off-the-shelf supervised detectors on C-HAT-Bench for document-level binary classification. To avoid truncation under detector input-length

Table 1: Detector performance under human–AI collaboration, measured by AUROC. The best two results are highlighted by 1st and 2nd. Here, MGT denotes the prefix-conditioned continuation condition; LC/HC indicate low/high coverage (15–35%/50–75%), and C/D indicate single-suffix/separated-span filling; Light/Deep denote expression polishing/structural rewriting; and L1–L3 indicate increasing prompt detail, with L3+Ref additionally using a same-domain reference.

Detector	MGT	Content Filling				Machine Rewriting		Prompt-based Generation			
		LC-C	LC-D	HC-C	HC-D	Light	Deep	L1	L2	L3	L3+Ref.
Zero-shot detectors											
Log-Likelihood	0.743	0.651	0.631	0.715	0.695	0.634	0.629	0.758	0.730	0.694	0.697
LogRank	0.746	0.644	0.624	0.709	0.689	0.623	0.616	0.762	0.727	0.688	0.691
Entropy	0.317	0.374	0.386	0.333	0.334	0.393	0.396	0.303	0.324	0.341	0.338
DetectLLM-LRR	0.712	0.578	0.560	0.635	0.610	0.529	0.516	0.723	0.661	0.606	0.608
Fast-DetectGPT	0.725	0.637	0.618	0.671	0.635	0.585	0.525	0.727	0.608	0.600	0.606
Binoculars	0.874	0.634	0.579	0.770	0.673	0.626	0.648	0.867	0.794	0.752	0.756
LastDE	0.835	0.681	0.664	0.743	0.747	0.671	0.715	0.848	0.829	0.773	0.769
LastDE++	0.859	0.707	0.669	0.798	0.749	0.697	0.683	0.857	0.801	0.752	0.755
SpecDetect	0.895	0.777	0.748	0.851	0.829	0.763	0.773	0.906	0.882	0.842	0.843
SpecDetect++	0.887	0.749	0.714	0.828	0.791	0.748	0.749	0.894	0.851	0.806	0.807
Supervised detectors											
RADAR	0.540	0.373	0.405	0.311	0.300	0.434	0.464	0.583	0.351	0.370	0.378
ReMoDetect	0.833	0.706	0.679	0.798	0.761	0.730	0.800	0.879	0.855	0.807	0.804
Longformer	0.826	0.546	0.514	0.578	0.531	0.522	0.542	0.877	0.683	0.649	0.644
BLOOMZ-3B	0.979	0.808	0.746	0.817	0.768	0.668	0.634	0.998	0.786	0.771	0.770
OpenAI RoBERTa	0.480	0.421	0.414	0.424	0.350	0.541	0.582	0.482	0.483	0.487	0.494
ConDA	0.501	0.503	0.487	0.526	0.468	0.471	0.479	0.557	0.638	0.587	0.581
ChatGPT Detector	0.724	0.599	0.545	0.724	0.640	0.652	0.694	0.741	0.736	0.686	0.693

limits, we use 512-token sliding windows and average window-level scores. We conduct a stride sensitivity analysis and adopt strides of 256 and 410 tokens for zero-shot and supervised detectors, respectively. Further details of the sliding-window hyperparameter analysis are provided in Appendix C.1. For each detector and condition, we report the mean AUROC, Aggregate results weight detectors and the ten collaboration conditions equally, with relative decreases measured against the corresponding mean MGT AUROC. with MGT as the reference condition. Additional experimental results are reported in Appendix D.1.

Experimental results. Table 1 reports the mean AUROC of zero-shot and supervised detectors for MGT and each fine-grained collaboration condition. Figure 3 visualizes the corresponding AUROC differences relative to MGT across the three collaboration subsets.

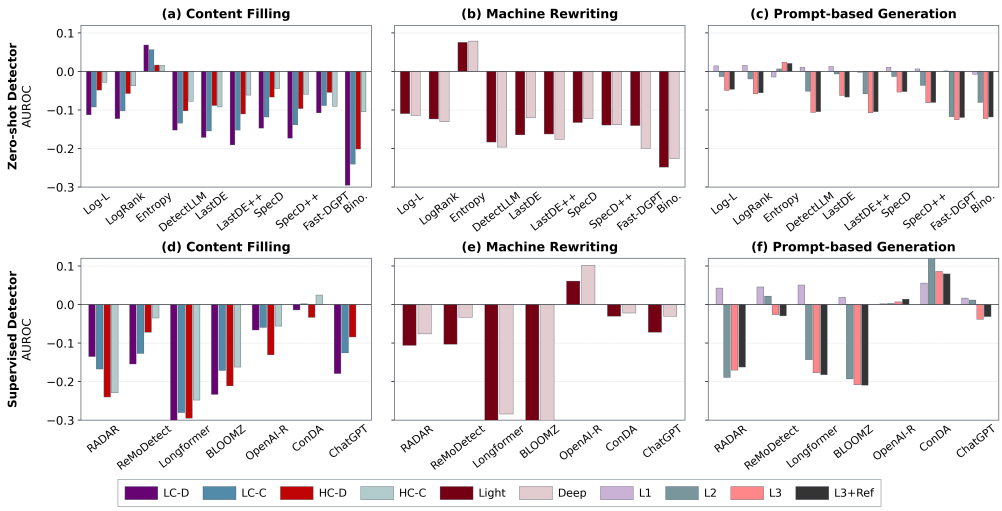

Figure 3: Detector performance relative to MGT across human–AI collaboration conditions. Top and bottom rows show zero-shot and supervised detectors, respectively.

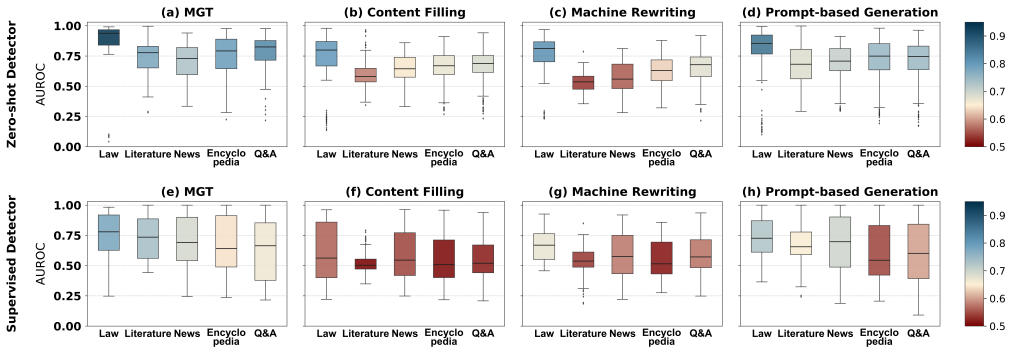

Figure 4: AUROC distributions across five text domains for zero-shot (top) and supervised (bottom) detectors on MGT and human–AI collaborative text. Box colors indicate median AUROC.

Observation ❶: Human–AI collaboration changes the detectability of machine involvement text. Relative to MGT, mean AUROC decreases by 11.1% and 13.3% for zero-shot and supervised detectors, respectively, when human–AI collaboration conditions are equally weighted (Table 1). Content Filling and Machine Rewriting reduce mean AUROC in both groups, while Prompt-based Generation shifts from performance close to MGT at L1 to lower performance at L3. Content mixing may dilute machine-generation cues at the document level, while rewriting and richer prompts may alter the lexical and stylistic patterns used for detection. The comparable results for L3 and L3+Ref. suggest that reference text has limited additional impact under already detailed prompt constraints. Thus, MGT performance alone cannot capture detection difficulty across collaboration conditions, which also depends on the form and extent of human intervention.

Observation ❷: Zero-shot detectors exhibit consistent responses to human–AI collaboration conditions. Within each collaboration subset, zero-shot detectors generally exhibit similar performance trends across human–AI collaboration conditions (Figure 3(a)–(c)). We attribute this consistency to their shared reliance on language-model statistics, including token probabilities, token rankings, and sequence-level distributional features. Collaboration may reduce the discriminative value of these shared cues, affecting multiple detectors despite differences in their scoring functions. Thus, diversity in scoring functions does not necessarily imply diversity in detection evidence; shared statistical dependencies can create common vulnerabilities to collaborative text.

Observation ❸: Supervised detectors exhibit method-dependent responses to human–AI collaboration conditions. As shown in Figure 3(d)–(f), supervised detectors exhibit different, sometimes opposing, performance changes under the same collaboration conditions. We attribute this variation to differences in training corpora, architectures, and learning objectives, which shape detector-specific decision boundaries. A collaboration-induced feature change may therefore weaken one detector's learned cues while preserving or reinforcing those used by another. The absence of a consistent difficulty ordering across the evaluated detectors suggests that detectability depends on the interaction between collaboration conditions and learned decision criteria.

4.2 Detection Sensitivity Across Text Domains (RQ2)

Experiment design. To assess the impact of text domains on detection performance, we compare zero-shot and supervised detectors across C-HAT-240k's five domains under MGT and three human–AI collaboration settings. Additional experimental results are provided in Appendix D.2.

Experimental results. Figure 4 presents the AUROC distributions of the two detector families across different text domains. Subfigures (a)–(d) correspond to zero-shot detectors, while Subfigures (e)–(h) correspond to supervised detectors. We derive the following key observations.

Observation ❹: Text domains exhibit stable but uneven detection difficulty. As shown in Figure 4, the law domain achieves the highest median AUROC in all eight subfigures, whereas literature and news generally yield lower detection performance, with this pattern remaining broadly stable across production settings. This consistency suggests that domain characteristics systematically shape the separability of human and machine text: the constrained terminology and discourse conventions of legal writing may expose generation artifacts, whereas the greater expressive variation in

open-ended domains can increase feature overlap. Thus, text domain constitutes a systematic source of variation in detection difficulty rather than a secondary evaluation factor.

Observation ❺: Zero-shot and supervised detectors exhibit distinct domain sensitivities. Zero-shot detectors preserve a stable domain ordering across collaboration settings, with an average IQR of 0.18, whereas supervised detectors show more dispersed and less consistent distributions, with the average IQR increasing to 0.31 and the weakest domain varying across settings. This difference reflects their underlying evidence: zero-shot detectors rely on shared probability and token-ranking signals, while supervised detectors learn decision boundaries shaped by their training corpora and architectures. Consequently, aggregate scores can conceal domain-specific weaknesses, making cross-domain coverage essential for assessing detector reliability.

4.3 Boundary Localization in Human–AI Collaborative Texts (RQ3)

Experiment design. We evaluate the fine-tuned WCP and the DeBERTa–BiGRU–CRF on the Content Filling subset. By aligning masking plans with the final texts, we recover machine-generated spans and derive sentence-level authorship labels and gold boundaries. We report sentence-level κ, WindowDiff, and Boundary F1@1, while additional preprocessing details and experimental results are provided in the appendix D.3.

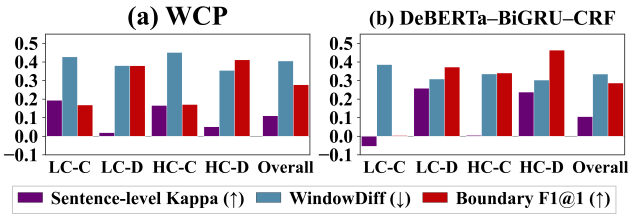

Figure 5: Boundary-localization performance of WCP and DeBERTa–BiGRU–CRF across Content Filling conditions.

Experimental results. Figure 5 reports sentence-level κ, WindowDiff, and Boundary F1@1 for WCP and the DeBERTa–BiGRU–CRF in panels (a) and (b), respectively.

Observation ❻: Boundary matching and authorship recovery respond differently to boundary density. Across coverage levels, WCP achieves higher Boundary F1@1 under dispersed filling but lower sentence-level κ than under contiguous filling. WCP proposes roughly four change points per text, whereas contiguous texts have one gold boundary and dispersed texts have about five. This mismatch makes contiguous texts prone to oversegmentation and partly explains higher boundary-matching scores under dispersed filling. Because WCP assigns sentence labels separately from change-point proposals, matching a boundary does not ensure authorship assignment on either side.

Observation ❼: Low-coverage contiguous filling exposes opposing failure modes. DeBERTa–BiGRU–CRF's Boundary F1@1 falls to 0.004, compared with 0.341 at high coverage; 70.9% of these samples yield no predicted boundary. As its token predictions are mapped to sentence-level labels, this pattern points to missed authorship transitions rather than small boundary offsets. WCP, in contrast, tends to predict excess boundaries on the same texts.

4.4 Cross-Domain Generalization of AIGC-detector-zhv2 (RQ4)

Experiment design. We train AIGC-detector-zhv2 separately on each of five domains and evaluate it across all five. Diagonal and off-diagonal cells denote in-domain and cross-domain tests. Each cell reports ACC using a Youden's J threshold selected from its labeled test samples; the results therefore reflect target-calibrated performance. Further details appear in Appendix D.4.

Experimental results. Figure 6 presents the resulting 5×5 ACC matrix for AIGC-detector-zhv2. Rows and columns denote the training and test domains, respectively; each row reflects source-domain transferability, while each column reflects target-domain difficulty.

Observation ❽: News-trained AIGC-detector-zhv2 exhibits the strongest cross-domain transferability. Training on News produces the highest mean cross-domain ACC (87.45%), with 86.1%–88.6% on the other domains, whereas Q&A yields 79.9% and falls to 61.8% on Law. This contrast indicates that transferability is shaped by the specificity of the learned cues, rather than by the magnitude of the domain shift alone. News may encourage more domain-agnostic cues through its

broader lexical and discourse variation, whereas Q&A may induce patterns tied to question–answer. Thus, the source domain should be treated as an explicit factor in cross-domain evaluation, since these structures that transfer poorly to formal domains such as Law. High in-domain accuracy alone does not establish transferable detection cues.

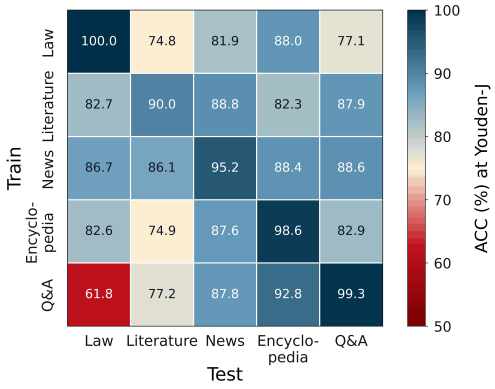

Figure 6: Cross-domain accuracies of AIGC-detector-zhv2 across five text domains.

Observation ❾: Literature is a target-domain bottleneck but a relatively transferable source domain. For AIGC-detector-zhv2, the mean cross-domain ACC is lowest when Literature is the target domain (78.24%), but reaches 85.43% when the detector is trained on Literature and evaluated on the remaining domains. This asymmetry shows that Literature is not uniformly incompatible with the other domains: incoming transfer is difficult, whereas outgoing transfer remains relatively strong. Its greater lexical and stylistic variation may obscure cues learned from other domains, while training on such variation may encourage a broader decision rule. Thus, cross-domain generalization is directional, and aggregate performance can conceal target-specific bottlenecks.

5 Conclusion and Future Directions

We present C-HAT-Bench, a benchmark for studying Chinese human–AI collaborative text detection. By linking shared human-written sources to machine-involved variants under controlled production settings, the benchmark enables detection behavior to be examined as collaboration granularity, source domain, and generative model vary. Our experiments show that detector performance is substantially affected by collaborative conditions and domain shifts, revealing limitations that are difficult to observe in evaluations based only on fully human-written and fully machine-generated text. We hope C-HAT-Bench will encourage the field to treat human–AI collaboration as a standard evaluation setting and promote the development of detection methods that remain reliable as writing processes and data distributions change.

These findings motivate three directions for future research:

- **Detection under human–AI collaboration.** A document that retains human-written context and contains only partial model intervention may not exhibit the same cues as fully machine-generated text. As the form and extent of collaboration change, the boundary between human and machine authorship may become less explicit, making collaborative text detection a fundamental unresolved problem.
- **Generalization beyond the evaluation condition.** Our generalization experiments show that detector performance deteriorates substantially on out-of-domain data, indicating that strong results under the original distribution do not guarantee reliable detection under other conditions. Improving generalization performance should therefore remain an important direction for future research, so that detector decisions remain dependable when the evaluation data differ from those used during development.
- **Sensitivity to collaboration versus false positives.** It is also possible that greater sensitivity to subtle machine involvement may increase false positives on high-quality human-written text. Such texts could contain regular or highly polished linguistic patterns that resemble machine-produced text even when no model has participated. Whether this trade-off exists, which types of human writing are most vulnerable, and how it varies across collaboration conditions remain open questions for future investigation.

Limitations. The current benchmark focuses on controlled fine-grained collaboration settings and does not yet examine more complex collaboration patterns or incorporate feedback loops into text data construction. We hope future research will extend this work toward richer collaboration processes and feedback-aware benchmark design.

AI USE STATEMENT

During manuscript preparation, generative AI tools were used only for limited language polishing and presentation-related assistance. They were not used to make scientific decisions, fabricate experimental results, or create citations. The authors reviewed the manuscript and take responsibility for the study design, analyses, claims, and final text.

ETHICS STATEMENT

This work uses existing Chinese text corpora and model-generated text variants and does not involve human-subject experiments. Source materials and released models are used subject to their applicable licenses and terms. The benchmark is intended for research and evaluation. Because detection performance can vary across domains, writing styles, and generation conditions, and because false positives may affect human-written text, detector outputs should be treated as supporting evidence and interpreted with human judgment rather than used as the sole basis for disciplinary or other consequential decisions.

REFERENCES

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. Fast-DetectGPT: Efficient Zero-Shot Detection of Machine-Generated Text via Conditional Probability Curvature. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *Transactions of the Association for Computational Linguistics*, 9:127–143, 2021.

Amrita Bhattacharjee, Tharindu Kumarage, Raha Moraffah, and Huan Liu. ConDA: Contrastive Domain Adaptation for AI-Generated Text Detection. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 598–610. Association for Computational Linguistics, 2023.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models Are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33*, pp. 1877–1901, 2020.

Jiaqi Chen, Xiaoye Zhu, Tianyang Liu, Ying Chen, Xinhui Chen, Yiwen Yuan, Chak Tou Leong, Zuchao Li, Long Tang, Lei Zhang, Chenyu Yan, Guanghao Mei, Jie Zhang, and Lefei Zhang. Imitate before Detect: Aligning Machine Stylistic Preference for Machine-Revised Text Detection. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence*, pp. 23559–23567. Association for the Advancement of Artificial Intelligence, 2025.

Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Joseph Cornelius, Oscar Lithgow-Serrano, Sandra Mitrovic, Ljiljana Dolamic, and Fabio Rinaldi. 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)*, pp. 8029–8057. Association for Computational Linguistics, 2024.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, Jamie Hayes, Nidhi Vyas, Majd Al Merey, Jonah Brown-Cohen, Rudy Bunel, Borja Balle, Taylan Cemgil, Zahra

Ahmed, Kitty Stacpoole, Ilia Shumailov, Ciprian Baetu, Sven Gowal, Demis Hassabis, and Pushmeet Kohli. Scalable watermarking for identifying large language model outputs. *Nature*, 634 (8035):818–823, 2024.

DeepSeek-AI. DeepSeek-V4: Towards Highly Efficient Million-Token Context Intelligence. arXiv preprint arXiv:2606.19348, 2026.

Liam Dugan, Alyssa Hwang, Filip Trhlík, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. RAID: A Shared Benchmark for Robust Evaluation of Machine-Generated Text Detectors. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12463–12492. Association for Computational Linguistics, 2024.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M. Rush. GLTR: Statistical Detection and Visualization of Generated Text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 111–116. Association for Computational Linguistics, 2019.

GLM-5 Team et al. GLM-5: From Vibe Coding to Agentic Engineering, 2026.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How Close Is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection. arXiv preprint arXiv:2301.07597, 2023.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Spotting LLMs with Binoculars: Zero-Shot Detection of Machine-Generated Text. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 17519–17537, 2024.

Wei Hao, Ran Li, Weiliang Zhao, Junfeng Yang, and Chengzhi Mao. Learning to rewrite: Generalized llm-generated text detection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 6421–6434, 2025.

Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. MGTBench: Benchmarking Machine-Generated Text Detection. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2251–2265. Association for Computing Machinery, 2024.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. RADAR: Robust AI-Text Detection via Adversarial Learning. In *Advances in Neural Information Processing Systems 36*, pp. 15077–15095, 2023.

Lei Jiang, Desheng Wu, and Xiaolong Zheng. Sendetex: Sentence-level ai-generated text detection for human-ai hybrid content via style and context fusion. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5287–5302, 2025.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 17061–17084, 2023.

Kristian Kuznetsov, Eduard Tulchinskii, Laida Kushnareva, German Magai, Serguei Barannikov, Sergey Nikolenko, and Irina Piontkovskaya. Robust ai-generated text detection by restricted embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 17036–17055, 2024.

Hyunseok Lee, Jihoon Tack, and Jinwoo Shin. ReMoDetect: Reward Models Recognize Aligned LLM's Generations. In *Advances in Neural Information Processing Systems 37*, pp. 2886–2913, 2024.

Mina Lee, Percy Liang, and Qian Yang. 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*, pp. 1–19, 2022.

Haitao Li, Yunqiu Shao, Yueyue Wu, Qingyao Ai, Yixiao Ma, and Yiqun Liu. LeCaRDv2: A Large-Scale Chinese Legal Case Retrieval Dataset. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2251–2260. Association for Computing Machinery, 2024a.

Mengchu Li, Jin Zhu, Jinglai Li, and Chengchun Shi. Segmenting Human–LLM Co-authored Text via Change Point Detection. In *The First Conference on Statistics and Trustworthy AI for Cross (X)-Domain Acceleration*, 2026.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. MAGE: Machine-Generated Text Detection in the Wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 36–53. Association for Computational Linguistics, 2024b.

Haitong Luo, Weiyao Zhang, Suhang Wang, Wenji Zou, Chungang Lin, Xuying Meng, and Yujun Zhang. SpecDetect: Simple, Fast, and Training-Free Detection of LLM-Generated Text via Spectral Analysis. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence*, pp. 32356–32364. Association for the Advancement of Artificial Intelligence, 2026.

Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, and Maria Bielikova. MULTITuDE: Large-Scale Multilingual Machine-Generated Text Detection Benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9960–9987. Association for Computational Linguistics, 2023.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. DetectGPT: Zero-Shot Machine-Generated Text Detection Using Probability Curvature. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 24950–24962, 2023.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual Generalization through Multitask Finetuning. In *Proceedings of the Eleventh International Conference on Learning Representations*, 2023.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying Large Language Models and Knowledge Graphs: A Roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599, 2024.

Lev Pevzner and Marti A. Hearst. A Critique and Improvement of an Evaluation Metric for Text Segmentation. *Computational Linguistics*, 28(1):19–36, 2002.

Chenxi Qing, Junxi Wu, Zheng Liu, Yixiang Qiu, Hongyao Yu, Bin Chen, Hao Wu, and Shu-Tao Xia. C-ReD: A Comprehensive Chinese Benchmark for AI-Generated Text Detection Derived from Real-World Prompts. In *Findings of the Association for Computational Linguistics: 64th Annual Meeting of the Association for Computational Linguistics*, pp. 42703–42733. Association for Computational Linguistics, 2026.

Gerrit Quaremba, Elizabeth Black, Denny Vrandečić, and Elena Simperl. TSM-Bench: Detecting LLM-Generated Text in Real-World Wikipedia Editing Practices. In *Proceedings of the Fourteenth International Conference on Learning Representations*, 2026.

L. D. M. S. Sai Teja, N. Siva Gopala Krishna, Ufaq Khan, Muhammad Haris Khan, and Atul Mishra. DAMASHA: Detecting AI in Mixed Adversarial Texts via Segmentation with Human-Interpretable Attribution. In *Findings of the Association for Computational Linguistics: 2026 Conference of the European Chapter of the Association for Computational Linguistics*, pp. 6189–6206. Association for Computational Linguistics, 2026.

Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, Miles McCain, Alex Newhouse, Jason Blazakis, Kris McGuffie, and Jasmine Wang. Release Strategies and the Social Impacts of Language Models. arXiv preprint arXiv:1908.09203, 2019.

Jinyan Su, Terry Yue Zhuo, Di Wang, and Preslav Nakov. DetectLLM: Leveraging Log-Rank Information for Zero-Shot Detection of Machine-Generated Text. In *Findings of the Association for Computational Linguistics: 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12395–12412. Association for Computational Linguistics, 2023.

Yuchuan Tian, Hanting Chen, Xutao Wang, Zheyuan Bai, Qinghua Zhang, Ruifeng Li, Chao Xu, and Yunhe Wang. Multiscale Positive-Unlabeled Detection of AI-Generated Texts. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

Irina Tolstykh, Aleksandra Tsybina, Sergey Yakubson, Aleksandr Gordeev, Vladimir Dokholyan, and Maksim Kuprashevich. Gigacheck: Detecting llm-generated content via object-centric span localization. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 4349–4364, 2026.

Yiu-Kei Tsang, Ming Yan, Jinger Pan, and Megan Yin Kan Chan. A corpus of Chinese word segmentation agreement. *Behavior Research Methods*, 57(1):25, 2025.

Adaku Uchendu, Zeyu Ma, Thai Le, Rui Zhang, and Dongwon Lee. TURINGBENCH: A Benchmark Environment for Turing Test in the Age of Neural Text Generation. In *Findings of the Association for Computational Linguistics: 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 2001–2016. Association for Computational Linguistics, 2021.

Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. Ghostbuster: Detecting Text Ghostwritten by Large Language Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1702–1717. Association for Computational Linguistics, 2024.

Peng Wang, Lijie Li, Kexin Ren, Bowen Jiang, Dong Zhang, and Xipeng Qiu. Seqxgpt: Sentence-level ai-generated text detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 1144–1156, 2023.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. M4GT-Bench: Evaluation Benchmark for Black-Box Machine-Generated Text Detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3964–3992. Association for Computational Linguistics, 2024a.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. M4: Multi-Generator, Multi-Domain, and Multi-Lingual Black-Box Machine-Generated Text Detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1369–1407. Association for Computational Linguistics, 2024b.

Junchao Wu, Runzhe Zhan, Derek Fai Wong, Shu Yang, Xinyi Yang, Yulin Yuan, and Lidia Sam Chao. DetectRL: Benchmarking LLM-Generated Text Detection in Real-World Scenarios. In *Advances in Neural Information Processing Systems 37*, pp. 100369–100401, 2024.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. A Survey on LLM-Generated Text Detection: Necessity, Methods, and Future Directions. *Computational Linguistics*, 51(1):275–338, 2025.

Yihuai Xu, Yongwei Wang, Yifei Bi, Huangsen Cao, Zhouhan Lin, Yu Zhao, and Fei Wu. Training-free LLM-generated Text Detection by Mining Token Probability Sequences. In *The Thirteenth International Conference on Learning Representations*, 2025.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238, 2005.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, et al. Qwen3 Technical Report, 2025.

Sha Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, Xiao Liu, Yukuo Cen, Xu Zou, Zhilin Yang, and Jie Tang. WuDaoCorpora: A super large-scale Chinese corpora for pre-training language models. *AI Open*, 2:65–68, 2021.

Zijie Zeng, Shiqi Liu, Lele Sha, Zhuang Li, Kaixun Yang, Sannyuya Liu, Dragan Gašević, and Guangliang Chen. Detecting AI-Generated Sentences in Human-AI Collaborative Hybrid Texts: Challenges, Strategies, and Insights. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 7545–7553, 2024.

Qihui Zhang, Chujie Gao, Dongping Chen, Yue Huang, Yixin Huang, Zhenyang Sun, Shilin Zhang, Weiye Li, Zhengyan Fu, Yao Wan, and Lichao Sun. LLM-as-a-Coauthor: Can Mixed Human-Written and Machine-Generated Text Be Detected? In *Findings of the Association for Computational Linguistics: 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 409–436. Association for Computational Linguistics, 2024.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Zican Dong, Yupeng Hou, Beichen Zhang, Yingqian Min, Junjie Zhang, Peiyu Liu, Xiaolei Wang, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Yiwen Hu, Jian-Yun Nie, and Ji-Rong Wen. A Survey of Large Language Models. *Frontiers of Computer Science*, 20(12):2012627, 2026.

Yuping Zhou and Nianwen Xue. The chinese discourse treebank: A chinese corpus annotated with discourse relations. *Language Resources and Evaluation*, 49(2):397–431, 2014.

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

APPENDIX

TABLE OF CONTENTS

A DATASET CONSTRUCTION 16
A.1 Human-Written Text Collection 16
A.2 Data Cleaning and Validation 17
A.3 Generative Models 17
A.4 Prompt Templates 18
A.5 Dataset Statistics 20
B DETECTOR DETAILS 21
C EXPERIMENTAL SETUP AND DATA PREPROCESSING 23
C.1 Sliding-Window Strategy and Stride Selection 23
C.2 Boundary Localization Data Processing 24
C.3 Cross-Condition Generalization Data Processing 24
D SUPPLEMENTARY EXPERIMENTAL RESULTS 24
D.1 Detection Performance at Fixed False-Positive Rates 24
D.2 Supplementary Analysis of Domain Sensitivity 26
D.3 Supplementary Results for Boundary Localization 27
D.4 Supplementary Cross-Condition Generalization Results 28
E REPRESENTATIVE DATA CONSTRUCTION EXAMPLES 29

A Dataset Construction

A.1 Human-Written Text Collection

To ensure consistency in corpus size and basic text-length distributions across domains, we select 1,000 human-written texts from each domain. Specifically, we adopt a length-centered sampling strategy: using approximately 1,250 characters as the target length, we first prioritize candidate texts close to this length and then gradually expand toward shorter and longer texts until 1,000 samples are obtained for each domain. All selected texts are further required to satisfy the Chinese-character proportion criterion, yielding 5,000 HWT samples in total.

News THUCNews is a Chinese news classification corpus compiled from historical articles collected through Sina News RSS feeds between 2005 and 2011. It contains approximately 740,000 articles spanning 14 topical categories. For C-HAT-240k, we select news texts from six categories: finance, education, technology, society, politics, and sports. To ensure that Chinese is the predominant language, we further exclude articles containing an excessive proportion of English-language content.

Law LeCaRDv2 is a large-scale Chinese legal case retrieval dataset comprising 800 query cases and 55,192 candidate cases drawn from approximately 4.3 million criminal case documents, with broad coverage of criminal charges (Li et al., 2024a). We use its candidate-case collection and extract the `qw` field containing the full text of each case document, rather than concatenating the separately provided fields for case facts, judicial reasoning, and judgment outcomes. Case numbers, dates, statutory provisions, and other domain-specific information are retained.

Q&A Zhihu-KOL is a Chinese question-answering corpus collected from Zhihu, comprising 1,006,218 question–answer pairs whose responses were authored by key opinion leader (KOL) users. We use a subset of 50,000 pairs as the candidate pool. Because the original corpus contains a substantial number of recently posted responses, we retain only the 25,833 pairs posted before 2020, as determined by the *answer_creation_time* field, to reduce the risk of machine-generated content entering the HWT corpus. Responses containing an excessive proportion of English-language content are further excluded. For HWT construction, the response in the *RESPONSE* field serves as the evaluation text, while the corresponding question in the *INSTRUCTION* field is retained to preserve the question–answer association and support data traceability; the question itself is not included in the detector input.

Encyclopedia The encyclopedia texts are drawn from the Chinese Wikipedia dataset on ModelScope, which is compiled from Chinese Wikipedia entries and contains approximately 2.87 million open-domain encyclopedia articles. The corpus covers a wide range of entry types, including people, events, organizations, locations, concepts, and knowledge-oriented topics. Compared with news, legal, and question-answering texts, Wikipedia articles place greater emphasis on objective exposition, conceptual explanation, and factual description, making them suitable for constructing the encyclopedia-domain HWT subset. We select entries with sufficiently complete body text that are primarily written in Chinese and satisfy the length requirement, and further remove samples containing an excessive proportion of English-language content.

Literature The literature texts are drawn from a locally curated collection of modern Chinese literary works. We select 102 modern literary works covering multiple genres, including novels, essays, prose writings, letters, and drama. Candidate passages are segmented according to natural paragraph boundaries and chapter structure, after which tables of contents, copyright information, empty paragraphs, and passages that do not satisfy the length requirement are removed. To prevent a small number of long works from dominating the subset, we control the number of samples drawn from each work and finally select 1,000 continuous passages from different works as the literature-domain HWT subset.

A.2 Data Cleaning and Validation

Table 2: HWT cleaning and validation rules for C-HAT-240K.

Stage	Criterion	Rule
Human-text selection	Length-based selection	Starting from 1250 characters, progressively expand the selection range toward shorter and longer texts until 1,000 samples are collected per domain.
Human-text selection	Chinese-character ratio	For news, law, Q&A, and encyclopedia, start at 90% and relax the threshold when insufficient samples are available, with a lower bound of 60%. The denominator excludes whitespace.
Format cleaning	Whitespace and markup	Normalize line endings, remove redundant whitespace and invisible characters, and strip HTML/Markdown markup, code fences, and recognizable task-description prefixes. Text lengths are computed after cleaning.

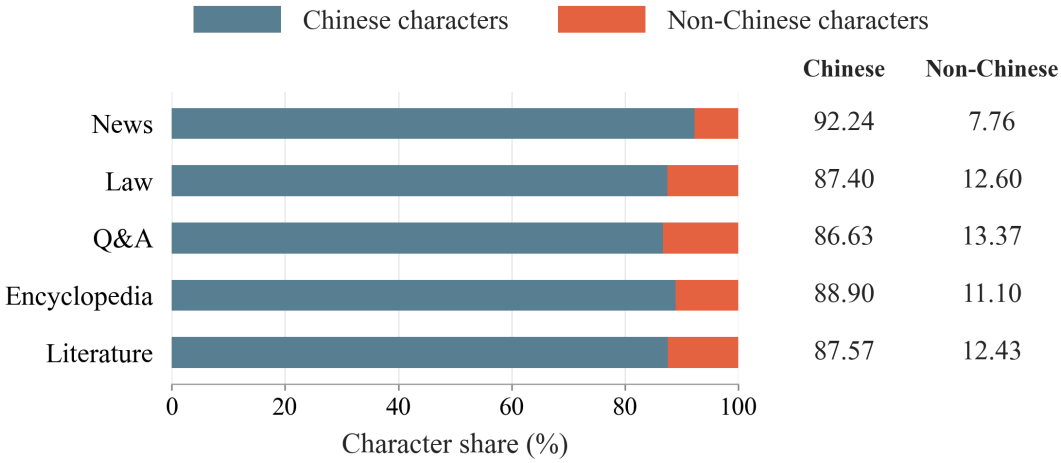

Figure 7: Character composition of the selected human corpus. Percentages are averaged over 1,000 texts per domain, excluding whitespace.

Table 2 summarizes the rules for human-text selection, generated-text cleaning, and record validation. Figure 7 shows that the selected human corpus is predominantly Chinese, with mean Chinese-character proportions ranging from 86.63% to 92.24% across the five domains. Non-Chinese characters include English letters, digits, and punctuation.

A.3 Generative Models

We use the following generative models to construct MGT and human–AI collaborative samples. The model names refer to the exact versions used in our experiments. Family-level technical reports are cited where available, while official documentation is provided in footnotes for the remaining models.

GPT-5.5 An OpenAI GPT-series model used to generate machine-involved samples in our benchmark.[5]

Qwen3.7-Plus A model in Alibaba's Qwen series used for Chinese and multilingual text generation, with its family-level technical background described in the Qwen3 technical report (Yang et al., 2025).

[5]OpenAI, *Models*, official model documentation. Available at: `https://platform.openai.com/docs/models` (accessed September 25, 2026).

GLM-5.2 A model in Z.ai's GLM series used for Chinese and English text generation, with its family-level technical background described in the GLM-5 technical report (GLM-5 Team et al., 2026).

Kimi K2.6 Model provided by Moonshot AI; the exact version is identified using the provider's official documentation. [6]

DeepSeek V4 Pro A model in the DeepSeek series used for text generation, with its technical background described in the DeepSeek-V4 technical report (DeepSeek-AI, 2026).

Gemini 3.5 Flash Gemini-series model provided by Google; the exact version is identified using the provider's official documentation. [7]

A.4 Prompt Templates

MGT

请根据下面这段文本开头进行续写。
要求：最终全文由"文本开头+ 续写内容"组成，全文长度应尽量接近原文长度。
原文长度：{human_text_length}
文本开头：{prefix_text}

Content filling

Continuous Span
请根据masked_text 中[MASK_1] 前面的原文，自然续写[MASK_1] 对应的内容，使生成后的文本与前文保持自然连贯。
生成要求：
1）只生成[MASK_1] 对应内容，不要改写masked_text 中已经存在的原文。
2）不要复述或重复masked_text 中已经存在的连续内容。
3）续写内容应贴合原文的主题、事实、语气、文体和叙述逻辑，并自然承接前文。
4）[MASK_1] 的generated_text 长度控制在min_generate_length 到max_generate_length 之间。

Dispersed Spans
请根据current_text 中当前MASK 所在位置的上下文，自然生成该MASK 对应的内容，使填补后的文本在上下文之间保持自然连贯。
生成要求：
1）只生成current_mask_id 对应内容，不要生成其他MASK，也不要改写current_text 中已经存在的原文。
2）不要复述或重复current_text 中已经存在的连续内容。
3）生成内容应贴合原文的主题、事实、语气、文体和上下文逻辑，并自然连接current_mask_id 所在位置的前后文。
4）generated_text 长度控制在min_generate_length 到max_generate_length 之间。
5）如果current_text 中已有前序MASK 的生成内容，请把这些已生成内容视为当前上下文的一部分，并在此基础上继续保持文本连贯。

Machine Rewriting

Light Revision
请对下面文本进行轻度表达润色。仅在局部层面优化用词、语序、衔接和句子流畅度，减少重复、生硬或不自然的表达。不要改变原文事实信息、核心含义、段落结构和整体写作风格，不要加入新的内容。只输出修改后的正文。

Deep Rewriting

[6] Moonshot AI, *Kimi API Documentation*, official platform documentation. Available at: https://platform.moonshot.cn/docs/intro (accessed September 25, 2026).

[7] Google, *Gemini API Documentation: Models*, official developer documentation. Available at: https://ai.google.dev/gemini-api/docs/models (accessed September 25, 2026).

请对下面文本进行深度改写。在保留原文事实信息、主题内容和领域文体特征的基础上，可以重新组织句式结构、表达方式和段落内部逻辑，使文本整体更加自然、清晰、连贯且具有重新写作后的表达形式。不要添加原文未包含的新事实。只输出改写后的正文。

Prompt-based Generation: News Templates

Level 1
请根据下面的新闻主题写一篇中文新闻报道，文本长度应尽量接近指定长度。
长度要求：{target_length}
新闻主题：{news_title}

Level 2
请写一篇新闻报道，主题是“{main_subject}”。文章需要围绕主要事件展开：{main_event}。事件结果或当前进展为：{result_or_status}。文本长度尽量接近{target_length}。

Level 3
你是一名擅长中文新闻报道写作的专业写作者。请根据以下任务大纲生成一篇新闻报道。正文应信息清楚、叙述完整、语言客观。只输出正文，文本长度尽量接近{target_length}。
任务大纲：
1）写作思路：全文应主要参考新闻报道的叙事顺序和信息重心，按照“导语概括核心事实—交代时间地点与相关主体—展开事件经过—说明结果进展—补充背景或来源信息”的方式组织内容。
2）新闻导语：围绕新闻主题“{main_subject}”写出开头，结合新闻类型{news_type} 和核心事件{main_event}，用简洁客观的语言交代最重要的信息。
3）基本情境：交代事件发生或报道涉及的时间{time}、地点{place} 和相关背景{background}；如果某项信息为空，不要自行补充。
4）相关主体：围绕涉及的人物或机构{people_or_orgs} 展开叙述，说明他们与事件之间的关系，不要添加资料之外的新主体。
5）事件经过：根据{event_process} 展开主体内容，使事件发展、前后因果和关键节点清楚连贯。
6）结果进展：根据{result_or_status} 说明事件结果、当前状态或后续进展，避免推断资料中没有给出的未来发展。
7）来源补充：结尾可结合{source_or_reporter} 和{commentary} 补充消息来源、记者信息或评论性内容；如果这些信息为空，则围绕事件结果自然收束全文。

Level 3+Ref
你是一名擅长中文新闻报道写作的专业写作者。请根据以下任务大纲生成一篇新闻报道。正文应信息清楚、叙述完整、语言客观。只输出正文，文本长度尽量接近{target_length}。
任务大纲同Level 3。
下面提供一篇同领域的参考文本，注意仅用作参考。正文的主要信息、内容安排和写作重点应以任务大纲为准。
参考文本：{reference_text}

This section presents the prompts used for the four text-production settings. The templates control the input information, generation scope, and content constraints, thereby defining different levels of machine involvement.

Machine-Generated Text. Only the opening segment of a human-written text and its original length are provided. The complete source text, topic description, and detailed writing instructions are omitted. The model continues the text from the given prefix while keeping the final length close to that of the source. This minimal prompt leaves content planning and expression largely to the model.

Content Filling. Within each domain, we divide the 1,000 source texts into three approximately equal groups, duplicate the groups, and assign the six resulting groups to the six generators, one group per generator. Each source text is thus processed by two generators under all four filling conditions. Separate prompts handle contiguous and dispersed spans. For contiguous spans, the model fills one mask from its preceding context while preserving the unmasked text and meeting the specified length constraint. For dispersed spans, it fills masks sequentially, receiving the updated text and current mask identifier at each step and generating only the targeted span. This avoids inconsistencies from filling multiple spans simultaneously and preserves coherence across generated segments.

Machine Rewriting. Machine Rewriting uses separate prompts for Light Rewriting and Deep Rewriting. Light Rewriting permits local changes to wording, word order, and sentence transitions while preserving the original facts, meaning, paragraph structure, and overall style. Deep Rewriting additionally permits changes to sentence structure and within-paragraph organization, while retaining the topic, main facts, and domain-specific style. Both prompts require the model to output only the rewritten text and prohibit the introduction of new facts.

Prompt-based Generation. Prompt-based Generation progressively increases the amount of structured information and writing constraints. Level 1 provides only the topic and target length. Level 2 adds key information, such as the main event and its outcome or current status. Level 3 further specifies the writing role, narrative organization, time, location, involved entities, event process, and source information. Level 3+Ref. additionally provides a same-domain reference text, but excludes the original source text of the current sample from reference selection. The reference guides content organization and domain-specific expression, while the generated text must follow the structured information of the current sample. News-domain templates are shown as examples; the other domains follow the same four-level hierarchy, with prompt content adapted to each domain.

A.5 Dataset Statistics

Table 3: Length statistics of human-written texts across five domains. Max/Min, Mean, SD, CV, and Count denote maximum/minimum character length, mean character length, standard deviation, coefficient of variation, and sample count, respectively.

Domain	Max	Min	Mean	SD	CV	Count
News	1707	1013	1296.22	157.88	12.18%	1000
Law	1500	1000	1254.71	146.36	11.66%	1000
Q&A	1487	960	1223.92	143.98	11.76%	1000
Encyclopedia	1532	988	1250.24	147.30	11.78%	1000
Literature	1505	1000	1249.82	144.56	11.57%	1000

Table 4: Length statistics of machine-involved samples across collaborative task conditions. Max/Min, Mean, SD, CV, and Count denote maximum/minimum character length, mean character length, standard deviation, coefficient of variation, and sample count, respectively.

Task	Condition	Max	Min	Mean	SD	CV	Count
AI Text	Overall	2062	611	1143.96	245.67	21.48%	28321
Content Filling	LC-C	1977	809	1274.13	178.74	14.03%	9855
Content Filling	LC-D	1998	704	1188.14	196.41	16.53%	9512
Content Filling	HC-C	2108	669	1220.80	219.61	17.99%	9336
Content Filling	HC-D	2092	646	1111.41	225.42	20.28%	8204
Content Filling	**Overall**	**2108**	**646**	**1202.31**	**212.89**	**17.71%**	**36907**
Machine Rewriting	Light	1925	733	1219.49	158.70	13.01%	29647
Machine Rewriting	Substantive	2106	616	1217.22	197.43	16.22%	29391
Machine Rewriting	**Overall**	**2106**	**616**	**1218.36**	**179.04**	**14.69%**	**59038**
Prompt-based Generation	Level 1	2133	625	1250.94	255.40	20.42%	28447
Prompt-based Generation	Level 2	2124	638	1343.53	204.37	15.21%	29341
Prompt-based Generation	Level 3	2096	625	1215.14	222.56	18.32%	29369
Prompt-based Generation	Level 3+Ref	2101	611	1191.56	222.93	18.71%	29116
Prompt-based Generation	**Overall**	**2133**	**611**	**1250.39**	**234.14**	**18.73%**	**116273**

The dataset combines balanced human-written sources with task-dependent variation in machine-involved samples.

Human-written sources. Table 3 contains 1,000 texts per domain. Their mean lengths range from 1,223.92 to 1,296.22 characters, with coefficients of variation between 11.57% and 12.18%.

Task-level variation. Table 4 shows that mean sample length ranges from 1,143.96 characters for MGT to 1,250.39 for Prompt-based Generation. The latter also contributes the most samples (116,273), indicating unequal task sizes.

Domain and generator coverage. The task–domain–generator breakdown shows that every production mode spans all five domains and six generators. Total sample counts per generator range from 37,957 (Kimi) to 41,268 (GPT).

Table 5: Counts of machine-involved samples by production type, domain, and generative model. GPT, Qwen, GLM, Kimi, DeepSeek, and Gemini denote GPT-5.5, Qwen3.7-Plus, GLM-5.2, Kimi K2.6, DeepSeek V4 Pro, and Gemini 3.5 Flash.

Task	Domain	GPT	Qwen	GLM	Kimi	DeepSeek	Gemini	Total
MGT	News	992	977	907	943	933	955	5707
	Law	999	978	918	828	863	991	5577
	Q&A	996	989	953	937	872	984	5731
	Encyclopedia	973	964	894	944	939	971	5685
	Literature	950	976	952	914	852	977	5621
MGT Total		4910	4884	4624	4566	4459	4878	**28321**
Content Filling	News	1320	1236	1234	1196	1237	1312	7535
	Law	1329	1208	1260	1156	1305	1314	7572
	Q&A	1319	1227	1273	1067	1210	1302	7398
	Encyclopedia	1310	1206	1260	1162	1223	1291	7452
	Literature	1298	1252	1208	678	1240	1274	6950
Content Infill Total		6576	6129	6235	5259	6215	6493	**36907**
Machine Rewriting	News	1940	1922	1930	1917	1948	1911	11568
	Law	2000	2000	2000	2000	1999	1996	11995
	Q&A	1968	1996	1988	1969	1994	1984	11899
	Encyclopedia	1996	1925	1926	1937	1926	1958	11668
	Literature	1993	1987	1991	1993	1985	1959	11908
Machine Rewriting Total		9897	9830	9835	9816	9852	9808	**59038**
Prompt-based Generation	News	4000	3962	3938	3207	3837	3908	22852
	Law	3978	3948	3982	3827	3659	3895	23289
	Q&A	3986	3957	3973	3935	3972	3978	23801
	Encyclopedia	3985	3794	3925	3885	3911	3975	23475
	Literature	3936	3979	3973	3462	3558	3948	22856
Prompt-based Generation Total		19885	19640	19791	18316	18937	19704	**116273**
Total		41268	40483	40485	37957	39463	40883	**240539**

B Detector Details

Zero-Shot Detectors

Log-Likelihood (Gehrmann et al., 2019) uses the mean log-likelihood assigned by a reference language model to the observed tokens as its detection score. The underlying assumption is that machine-generated text tends to conform more closely to the probability distribution learned by the generator and may therefore receive a higher likelihood.

LogRank (Gehrmann et al., 2019) performs detection using the average rank of the observed tokens in the language model's predictive distributions. Because generation procedures tend to select highly ranked candidate tokens, their rank distributions may differ from those of human-written text.

Entropy (Gehrmann et al., 2019) measures the predictive uncertainty of a language model at each token position and aggregates these values into a document-level score. This signal captures dif-

ferences in the concentration of the model's predictive distributions between human-written and machine-generated texts.

DetectLLM-LRR (Su et al., 2023) combines text likelihood with token log-rank through a likelihood–log-rank ratio. This criterion jointly captures the model's absolute predictive confidence and the relative positions of the observed tokens within its candidate distributions.

LastDE (Xu et al., 2025) treats the token-probability sequence as a time series and uses multiscale diversity entropy to characterize its local dynamics. These features are combined with global likelihood to distinguish the probability variation patterns of human-written and machine-generated texts.

LastDE++ (Xu et al., 2025) augments these dynamic features with efficient sampling and score normalization. Its final criterion compares the score of the observed text with those of contrast samples, improving the stability of the resulting predictions.

SpecDetect (Luo et al., 2026) analyzes token log-probability sequences in the frequency domain and uses the total spectral energy obtained through the discrete Fourier transform as its detection signal. This feature reflects differences in the magnitude and frequency structure of probability fluctuations.

SpecDetect++ (Luo et al., 2026) further introduces a sampling-based discrepancy mechanism. It compares the spectral energy of the observed text with the distribution derived from model-generated contrast samples and uses the standardized difference as the detection score.

Fast-DetectGPT (Bao et al., 2024) relies on conditional probability curvature, comparing the conditional probabilities of observed tokens with those of model-sampled alternatives. It thereby approximates the curvature-based criterion of DetectGPT at substantially lower computational cost.

Binoculars (Hans et al., 2024) uses an observer model and a performer model to compute perplexity and cross-perplexity, respectively, and determines text authorship from their ratio. Detection is therefore based on the relative confidence exhibited by two language models on the same text.

Supervised Detectors

RADAR (Hu et al., 2023) improves detection robustness through adversarial training between a paraphraser and a detector. The paraphraser rewrites machine-generated text to evade detection, while the detector learns to identify both the original and rewritten outputs. We directly use the publicly released detector checkpoint.

ReMoDetect (Lee et al., 2024) builds on reward modeling and employs experience replay to mitigate overfitting during continual preference fine-tuning. It further uses mixed-preference data created through partial machine rewriting to refine the classification boundary. We perform inference with its publicly released DeBERTa-based checkpoint.

Longformer (Li et al., 2024b) denotes the long-document detector released with MAGE. It is trained on human-written texts and machine-generated texts from multiple generators. We use the publicly available checkpoint.

BLOOMZ-3B (Muennighoff et al., 2023) is a binary classifier built on the multilingual instruction-tuned BLOOMZ-3B model and trained on a mixture of human-written and machine-generated texts. We use the publicly released mixed-detector checkpoint.

OpenAI RoBERTa (Solaiman et al., 2019) predicts text authorship using a RoBERTa classifier fine-tuned on human-written WebText documents and GPT-2 outputs. Although it was trained primarily on outputs from an earlier generation of language models, it provides a useful baseline for evaluating cross-model and cross-lingual transfer.

ConDA (Bhattacharjee et al., 2023) combines contrastive learning with unsupervised domain adaptation. Using labeled source-domain data and unlabeled target-domain data, it learns representations that are both class-discriminative and domain-invariant. We use its publicly released RoBERTa-based detector checkpoint.

ChatGPT Detector (Guo et al., 2023) is a RoBERTa classifier fine-tuned on paired human and ChatGPT answers from HC3. We adopt its publicly released Chinese checkpoint to examine how a detector trained on Chinese question-answering data transfers to collaborative texts from multiple domains.

AIGC-detector-zhv2 (Tian et al., 2024) is built on Chinese RoBERTa-wwm-ext and trained using a multiscale positive–unlabeled learning framework for Chinese machine-generated text detection. In the cross-condition experiments, we initialize the detector with its public weights and train it separately on the data from each source condition.

IMBD (Chen et al., 2025) first applies Style Preference Optimization to align a scoring model with machine-writing style preferences and then derives its detection score from Style-Conditional Probability Curvature. Following the original procedure, we train the model separately under each source condition to evaluate transfer across domains, collaborative tasks, and generative models.

Boundary Localization Detectors

WCP (Li et al., 2026) approaches boundary localization in human–AI mixed texts from a change-point detection perspective. It converts each sentence into a scalar detection score, forming a one-dimensional sequence, and then applies a weighted change-point detection algorithm to identify shifts in the score distribution. We use its weighted variant for sentence-level boundary prediction.

DeBERTa–BiGRU–CRF. We implement a token-level detector with a DeBERTa-v3-base encoder, a three-layer bidirectional GRU, and a CRF decoder to assign human- or machine-authorship labels to each token. A DeBERTa–BiGRU–CRF configuration has also been evaluated in DAMASHA (Sai Teja et al., 2026). We merge consecutive machine-labeled tokens into character-level spans and map them to the shared sentence-level representation for evaluation.

C Experimental Setup and Data Preprocessing

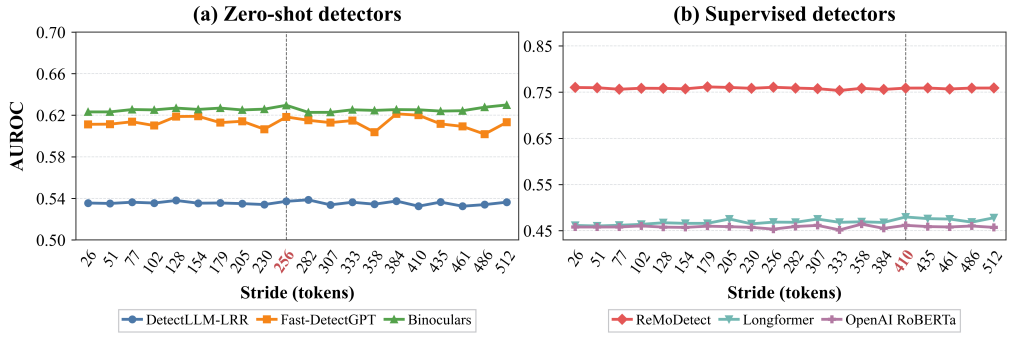

Figure 8: stride-sensitivity analysis for representative zero-shot and supervised detectors. The left panel shows zero-shot detectors, and the right panel shows supervised detectors. The highlighted x-axis value indicates the selected stride for the corresponding experiment.

C.1 Sliding-Window Strategy and Stride Selection

Windowing strategy. Each text is tokenized with the corresponding detector's tokenizer and divided into 512-token windows with stride s. If the regular windows leave the text ending uncovered, we include an additional end-aligned window. Windows are scored independently, and their scores are averaged to obtain the document-level score.

Stride sensitivity. We use 973 GPT-generated texts from the News domain of the Machine Rewriting task and their 973 matched human-written sources. With 512-token windows, we vary the overlap ratio from 0% to 95% in increments of five percentage points and compare the resulting strides using document-level AUROC.

Stride selection. Figure 8 shows that mean AUROC peaks at stride 256 across DetectLLM-LRR (Su et al., 2023), Fast-DetectGPT (Bao et al., 2024), and Binoculars (Hans et al., 2024) (left). For ReMoDetect (Lee et al., 2024), Longformer (Beltagy et al., 2021), and OpenAI RoBERTa (Solaiman et al., 2019), the mean peaks at stride 410 (right). We therefore adopt 256- and 410-token strides for zero-shot and supervised detectors, respectively. AUROC varies little across adjacent strides in this setting.

C.2 Boundary Localization Data Processing

We construct sentence-level boundary-localization instances by deriving gold machine-generated intervals, aligning detector outputs to the sentence level, and evaluating all methods under source-disjoint data splits.

Gold interval construction. We use the content-filling subset from Experiment 1. For each sample, the original text containing `[MASK]` markers is aligned with its corresponding generated text, and the spans replacing the masks are treated as gold machine-generated intervals. Samples with unreliable alignment are excluded.

Sentence-level annotation. Each text is segmented according to Chinese and English sentence-final punctuation. A sentence is labeled as machine-generated if it overlaps with at least one gold interval; otherwise, it is labeled as human-written. The gold boundaries are defined at positions where the labels of adjacent sentences change.

Prediction alignment. To place all detector outputs at the same granularity, WCP predicts change points directly over sentence sequences. The token-level predictions produced by DeBERTa–BiGRU–CRF are merged into character-level intervals and then mapped to sentence labels and boundary positions.

Data splitting and evaluation. Samples are grouped by domain and source-text ID before being divided into training, validation, and test sets, ensuring that variants derived from the same source text do not appear across different splits. We conduct the final evaluation on the test set using Cohen's κ, WindowDiff, and Boundary $F_1@1$.

C.3 Cross-Condition Generalization Data Processing

Cross-condition generalization. Experiment 4 evaluates how supervised detectors transfer when the training and test conditions differ.

- *Generalization axes.* We construct train–test matrices separately along the domain, collaborative-task, and generative-model dimensions, resulting in 5×5, 4×4, and 6×6 matrices, respectively.
- *Training and labels.* For each matrix row, one condition is used for training and every condition along the same dimension is used for testing. Detectors are trained on the corresponding training split and checkpointed using the validation split from that condition. Human-written texts form the negative class, whereas machine-involved texts form the positive class.
- *Threshold selection.* For each train–test cell, we select the decision threshold that maximizes Youden's J on its labeled evaluation samples.
- *Evaluation.* We report Accuracy at the cell-specific Youden threshold. Scores are macro-averaged after source-ID pairing and fine-grained condition grouping to reduce the influence of unequal sample counts across matrix cells.

D Supplementary Experimental Results

D.1 Detection Performance at Fixed False-Positive Rates

A strict false-positive budget makes collaborative text selectively invisible. Table 6 shows that at 1% FPR, the strongest detectors retain relatively high ACC and F1 on MGT and L1, whereas content filling, machine rewriting, and more complex prompt conditions generally produce lower scores. The degradation is more pronounced in F1 than in ACC, indicating that the strict operating point suppresses true-positive coverage even when the overall accuracy remains close to chance. This pattern suggests that shallow collaboration can preserve detectable machine regularities, while deeper human intervention or stronger generation constraints weaken the evidence available to the detector.

Increasing the FPR buys detection coverage, but not recovered evidence. Relaxing the FPR from 1% to 5% and 10% generally improves both ACC and F1, particularly for collaborative conditions that perform poorly under the strictest threshold. However, the relative difficulty across collaboration conditions remains uneven, and the strongest performance does not converge to a common

Table 6: Detector performance under human–AI collaboration, measured by ACC and F1 at 1%, 5%, and 10% FPR. The best two results are highlighted by 1st and 2nd. Values are reported as percentages and macro-averaged over 30 domain–generation-model combinations.

Detector	MGT		Content Filling								Machine Rewriting				Prompt-based Generation							
			LC-C		LC-D		HC-C		HC-D		Light		Subst.		L1		L2		L3		L3+Ref.	
	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
1% FPR *Zero-shot detectors*																						
Log-Likelihood	58.5	22.5	52.6	10.4	52.3	9.6	53.8	14.1	54.8	15.7	52.5	10.0	52.6	10.3	56.7	18.9	53.4	12.2	53.6	12.7	53.9	13.5
LogRank	58.4	21.5	52.4	9.6	52.3	9.1	53.4	12.8	54.5	14.8	52.2	9.0	52.4	9.4	56.9	18.5	53.2	11.4	53.4	11.8	53.7	12.4
Entropy	50.1	2.3	49.9	1.7	49.9	1.6	49.9	1.4	50.0	1.7	49.9	1.4	50.0	1.7	50.0	1.7	49.8	1.1	50.0	1.7	50.0	1.6
DetectLLM-LRR	53.2	10.7	50.4	3.2	50.4	3.3	50.6	3.8	50.6	4.0	50.2	2.5	50.2	2.5	53.4	10.8	50.6	3.7	50.7	4.3	50.8	4.4
Fast-DetectGPT	54.9	18.5	51.5	7.4	51.2	6.4	52.1	9.6	52.1	9.0	50.7	4.7	50.2	2.9	54.1	15.7	51.0	5.5	50.8	4.8	50.9	5.2
Binoculars	69.2	52.3	51.5	7.4	50.7	4.6	58.1	26.7	53.7	14.8	50.8	5.1	51.0	5.5	65.4	44.6	58.1	26.3	55.1	18.2	55.4	19.4
LastDE	53.9	14.9	50.5	3.7	50.4	3.6	50.9	5.1	51.1	6.1	50.4	3.2	50.8	4.6	54.1	14.7	52.7	10.6	51.7	7.6	51.8	7.9
LastDE++	60.0	32.0	51.5	7.6	51.0	5.9	54.4	17.1	52.5	11.2	50.9	5.2	50.8	4.7	57.3	25.0	53.8	14.0	52.0	9.1	52.0	9.1
SpecDetect	66.3	45.4	56.4	21.7	55.8	19.9	60.5	33.0	59.8	29.9	56.8	22.1	57.2	23.2	65.1	42.5	60.3	32.4	58.4	27.1	58.6	27.6
SpecDetect++	61.9	37.2	52.0	9.4	51.5	7.6	55.2	19.6	53.3	13.8	51.7	8.2	51.5	7.5	59.7	32.0	55.5	19.7	53.2	13.3	53.3	13.6
1% FPR *Supervised detectors*																						
RADAR	49.9	1.4	49.7	0.6	49.7	1.0	49.6	0.4	49.6	0.5	49.6	0.4	49.6	0.3	49.6	0.5	49.6	0.2	49.5	0.0	49.5	0.0
ReMoDetect	59.5	28.4	52.2	9.6	51.1	6.0	57.1	23.2	53.6	13.8	51.9	8.7	53.9	14.8	60.2	29.3	60.5	28.9	59.5	26.3	59.4	26.2
Longformer	55.6	19.9	49.8	1.4	50.0	1.8	49.8	1.3	49.8	1.3	49.8	1.0	50.1	2.4	56.7	23.0	51.4	6.6	50.7	4.6	50.8	4.8
BLOOMZ-3B	92.8	90.4	52.7	11.4	51.9	8.8	52.8	11.8	52.5	10.8	50.6	4.3	51.3	6.7	98.2	98.0	57.2	24.7	56.9	22.4	57.4	23.3
OpenAI RoBERTa	49.8	1.1	49.8	1.2	49.9	1.5	49.8	1.1	49.7	0.8	50.1	2.3	50.2	2.5	49.9	1.3	49.7	0.8	49.7	0.7	49.7	0.8
ConDA	49.8	1.1	49.8	1.3	49.9	1.5	49.8	1.1	49.8	1.1	49.8	1.0	49.8	0.9	49.9	1.4	49.9	1.6	49.8	1.2	49.8	1.2
ChatGPT Detector	51.9	8.3	50.3	2.8	50.2	2.5	50.9	4.9	50.7	4.2	50.4	3.1	50.3	2.7	51.2	6.1	50.6	4.0	51.2	6.0	51.4	6.4
5% FPR *Zero-shot detectors*																						
Log-Likelihood	60.8	30.3	56.1	22.2	55.7	21.2	58.3	27.6	58.4	26.5	56.3	22.1	56.5	22.4	60.1	28.4	57.8	24.6	57.9	24.7	58.1	25.1
LogRank	60.0	27.9	55.3	19.8	55.0	18.8	57.2	24.5	57.2	23.3	55.2	19.2	55.5	19.4	59.2	25.7	56.7	21.4	56.7	21.1	57.0	21.7
Entropy	49.2	5.6	49.2	5.7	49.3	6.1	48.8	4.5	49.2	5.2	49.3	6.0	49.8	7.4	49.0	4.6	48.8	4.0	49.3	5.4	49.2	5.3
DetectLLM-LRR	55.3	17.4	51.1	9.5	51.1	9.6	51.6	11.0	51.8	11.1	50.5	8.3	50.6	8.4	55.2	16.4	51.5	10.2	51.6	10.2	51.7	10.5
Fast-DetectGPT	60.5	37.9	54.4	22.1	53.7	20.4	56.0	26.6	55.0	23.0	52.5	16.6	51.0	11.9	59.8	35.3	53.1	17.9	52.5	16.7	52.9	17.7
Binoculars	76.9	69.5	54.0	21.4	52.3	16.2	64.2	45.6	58.3	32.1	52.9	18.3	53.5	19.9	74.1	65.1	65.3	48.4	60.9	38.6	61.5	39.9
LastDE	61.4	38.4	52.2	15.3	52.0	14.4	54.1	20.8	55.1	23.6	52.2	15.2	54.0	20.1	61.8	38.1	59.7	32.6	56.8	25.5	56.9	25.7
LastDE++	69.7	57.6	55.3	25.2	53.7	20.7	61.6	41.0	58.2	32.9	54.2	22.3	53.7	20.1	67.1	52.7	60.7	38.0	56.8	28.8	57.1	29.6
SpecDetect	74.7	65.4	63.3	41.4	61.5	37.2	68.9	54.7	67.1	50.1	63.0	40.0	63.4	40.6	74.8	65.5	70.9	58.2	66.6	48.6	66.9	49.3
SpecDetect++	72.4	62.9	57.2	30.5	55.4	25.6	63.7	46.0	60.3	38.4	56.5	28.7	56.2	27.5	71.7	61.3	64.8	47.5	60.6	38.2	61.0	39.1
5% FPR *Supervised detectors*																						
RADAR	49.9	8.1	48.8	4.3	49.3	6.1	48.0	1.9	48.4	3.4	48.5	3.4	48.5	3.6	50.2	8.8	48.1	2.2	47.7	0.9	47.8	1.0
ReMoDetect	70.9	55.1	58.8	31.7	56.7	27.3	66.8	48.0	62.8	40.5	59.1	33.0	63.6	42.4	71.5	55.5	70.1	49.8	69.2	47.1	69.0	46.8
Longformer	67.3	51.7	50.2	9.6	50.1	9.0	50.4	10.2	50.1	9.4	49.9	8.4	50.7	11.1	71.3	59.8	55.8	25.3	54.4	21.6	54.8	22.4
BLOOMZ-3B	91.6	89.6	57.5	29.4	55.3	23.9	58.2	30.9	56.4	26.4	51.6	13.8	51.4	13.6	96.5	96.6	59.5	35.1	58.4	31.4	58.8	32.1
OpenAI RoBERTa	48.9	4.9	49.1	5.2	49.3	6.0	48.9	4.6	48.5	3.7	50.4	9.4	50.7	10.4	49.0	5.1	48.5	3.4	48.4	3.2	48.5	3.6
ConDA	49.7	7.8	49.7	7.6	49.7	7.5	49.6	7.5	48.9	5.3	48.7	4.5	48.7	4.3	50.4	9.8	51.7	14.0	50.3	9.6	50.2	9.4
ChatGPT Detector	53.7	17.4	51.3	11.2	50.5	9.0	54.4	19.1	51.7	12.5	51.3	10.8	51.2	10.5	53.1	15.8	52.9	15.5	54.2	18.1	54.6	18.9
10% FPR *Zero-shot detectors*																						
Log-Likelihood	60.1	33.3	56.8	27.1	56.2	25.6	58.9	32.2	58.6	30.6	56.7	26.5	57.1	27.0	59.6	32.1	58.1	28.7	58.1	28.6	58.4	29.3
LogRank	59.2	30.8	55.8	24.2	55.4	23.2	57.9	29.3	57.4	27.1	55.7	23.5	55.8	23.5	58.6	29.0	56.9	25.5	56.9	25.1	57.1	25.5
Entropy	48.1	8.9	48.5	10.5	48.8	11.2	47.6	7.6	48.2	8.9	48.8	11.3	49.6	13.1	47.8	7.4	47.5	6.6	48.5	9.1	48.5	9.1
DetectLLM-LRR	55.5	21.4	51.7	14.3	51.5	13.7	52.2	15.9	52.0	14.9	50.8	12.6	50.9	12.5	55.0	19.4	52.0	14.4	51.9	13.8	51.9	14.0
Fast-DetectGPT	63.9	49.1	56.5	33.1	55.8	31.5	58.5	37.8	57.0	33.2	53.9	26.4	51.6	19.9	63.1	46.7	55.0	28.5	54.3	27.2	54.6	28.1
Binoculars	79.8	76.1	56.4	33.6	53.8	26.7	68.0	57.1	60.8	42.4	55.2	30.4	56.3	32.9	77.9	73.5	69.7	60.3	65.0	51.2	65.6	52.4
LastDE	66.2	52.1	54.6	27.0	54.1	25.9	57.9	35.3	58.9	37.4	54.4	27.0	57.2	33.3	66.6	52.2	64.3	47.1	60.5	38.6	60.3	38.3
LastDE++	74.9	69.4	59.0	39.9	57.0	35.2	66.9	56.1	62.7	48.0	58.0	37.9	57.0	34.6	73.4	67.2	66.2	54.2	61.6	44.8	62.0	45.9
SpecDetect	77.6	72.7	65.7	50.7	63.5	45.7	71.7	63.1	69.6	58.6	64.7	48.0	65.3	48.6	78.4	73.8	75.1	68.2	70.2	59.1	70.3	59.4
SpecDetect++	77.0	72.7	60.9	43.9	58.5	38.5	68.4	58.8	64.7	52.0	60.5	43.5	60.1	42.0	76.9	72.2	70.3	61.3	65.5	52.2	65.9	53.1
10% FPR *Supervised detectors*																						
RADAR	50.4	16.5	47.8	8.8	48.7	11.6	46.5	4.8	47.1	7.5	47.5	8.1	47.8	9.2	52.1	18.9	46.6	5.3	45.9	3.2	45.9	3.3
ReMoDetect	74.4	62.9	62.6	42.9	60.7	39.5	70.8	57.2	67.0	51.2	63.4	44.8	68.2	53.5	75.4	64.9	72.4	57.1	71.2	53.7	71.0	53.7
Longformer	71.5	62.6	50.6	17.7	49.9	15.5	51.2	19.7	50.1	16.8	49.8	15.3	50.8	18.3	76.2	70.7	58.2	35.8	56.3	31.5	56.6	31.9
BLOOMZ-3B	89.7	88.6	61.9	43.5	58.3	35.6	62.4	44.5	59.9	38.9	52.8	22.7	51.8	21.4	94.3	95.0	61.6	44.1	59.8	39.5	60.2	40.0
OpenAI RoBERTa	48.1	10.1	48.4	10.0	48.7	10.9	47.8	8.4	47.0	6.8	50.8	16.6	51.5	18.4	48.2	10.0	47.3	7.7	47.5	7.9	47.7	8.4
ConDA	49.8	14.9	49.5	14.4	49.4	13.9	49.8	15.5	48.5	11.6	47.8	9.1	47.6	8.8	51.0	18.0	53.8	25.4	51.2	19.2	50.9	18.2
ChatGPT Detector	55.5	26.7	52.6	19.1	51.0	15.6	56.7	29.0	53.7	22.7	52.9	19.9	53.2	21.2	55.3	26.2	56.7	29.1	56.9	27.3	57.0	27.2

ordering. Increasing the FPR changes the decision threshold and allows weaker machine-generation scores to be accepted, but it cannot restore signals that collaboration has removed or redistributed. Thus, a more permissive operating point trades false positives for coverage rather than resolving the underlying distribution shift.

Supervision redistributes detector sensitivity rather than removing distribution shift. The upper-performing zero-shot detectors maintain a relatively coherent performance pattern across FPR levels, whereas supervised detectors exhibit condition-dependent crossovers. BLOOMZ-3B performs particularly well on MGT and L1, but loses this advantage on many collaborative conditions, where ReMoDetect becomes more competitive. This contrast indicates that supervision does not produce a collaboration-invariant decision boundary; instead, it transfers the training distribution's preferences into the detector. The decisive factor is therefore not whether a detector is supervised, but whether its learned or intrinsic evidence remains aligned with the collaboration process under evaluation.

D.2 Supplementary Analysis of Domain Sensitivity

Table 7: Domain-wise AUROC of zero-shot and off-the-shelf supervised detectors. Each value is macro-averaged over six generation models and the fine-grained conditions within each of the four text-production settings, followed by an equal-weight average over the four settings. Within each detector family and domain, the best and second-best results are shown in boldface and underlined, respectively.

Detector	Law	Literature	News	Encyclopedia	Q&A
Zero-shot detectors					
Log-Likelihood	0.891	0.575	0.618	0.676	0.698
LogRank	0.873	0.583	0.614	0.669	0.698
Entropy	0.182	0.479	0.399	0.346	0.337
DetectLLM-LRR	0.663	0.602	0.572	0.602	0.661
Fast-DetectGPT	0.783	0.602	0.552	0.604	0.655
Binoculars	0.801	**0.809**	0.662	0.736	0.702
LastDE	0.832	<u>0.686</u>	0.766	0.768	0.749
LastDE++	0.876	0.685	0.728	0.761	0.788
SpecDetect	**0.958**	0.675	<u>0.790</u>	**0.875**	**0.868**
SpecDetect++	<u>0.905</u>	0.682	**0.794**	<u>0.841</u>	<u>0.836</u>
Supervised detectors					
RADAR	0.529	0.580	0.397	0.421	0.268
ReMoDetect	**0.923**	0.627	<u>0.870</u>	<u>0.854</u>	0.690
Longformer	0.663	**0.698**	0.686	0.553	0.667
BLOOMZ-3B	<u>0.837</u>	0.594	**0.894**	**0.874**	**0.858**
OpenAI RoBERTa	0.508	0.572	0.502	0.390	0.441
ConDA	0.647	0.572	0.439	0.450	0.470
ChatGPT Detector	0.759	<u>0.653</u>	0.672	0.621	<u>0.719</u>

Domain variation creates a structured but uneven detection landscape. Table 7 summarizes domain-wise AUROC after averaging over generation models and fine-grained collaboration conditions. Law is generally the most detectable domain, whereas Literature and News tend to yield lower scores. This pattern indicates that domain characteristics systematically affect the separability between human-written and machine-generated text: the constrained vocabulary and discourse structure of legal text may expose generation artifacts more consistently, while the broader stylistic variation of open-ended domains increases the overlap between the two classes.

Zero-shot detectors exhibit relatively stable domain sensitivity because they rely on shared statistical evidence. Their domain ordering remains broadly consistent across text-production settings, with SpecDetect and SpecDetect++ achieving strong overall performance and Binoculars showing a notable advantage on Literature. Although their scoring functions differ, these detectors primarily exploit probability distributions, token ranks, or related distributional statistics derived from lan-

guage models. Domain-specific regularities therefore modulate a common type of evidence: constrained legal writing can amplify deviations in model-based statistics, whereas more stylistically diverse domains weaken their consistency.

Supervised detectors transform domain shifts into model-specific weaknesses. The strongest detector changes across domains: ReMoDetect performs best on Law, BLOOMZ-3B dominates News, Encyclopedia, and Q&A, while Longformer is strongest on Literature. This variation reflects the dependence of supervised decision boundaries on training data, model architecture, and optimization objectives. Consequently, detector reliability cannot be inferred from an aggregate AUROC alone; it depends on whether the learned cues remain aligned with the target domain.

Table 8: Detailed boundary-localization performance of the two fine-tuned detectors on the common content-filling test set. Here, n denotes the number of evaluated documents, and values are document-level means $\pm$ population standard deviations. Boundary F1@1 uses one-to-one matching between predicted and gold boundaries within a tolerance of one sentence; arrows indicate the preferred direction.

Condition	Detector	n	Sentence $\kappa \uparrow$	WindowDiff $\downarrow$	Boundary F1@1 $\uparrow$
LC-C	WCP	1962	0.194 ± 0.343	0.428 ± 0.206	0.168 ± 0.226
LC-C	DeBERTa–BiGRU–CRF	1962	-0.054 ± 0.101	0.386 ± 0.184	0.004 ± 0.051
LC-D	WCP	1886	0.019 ± 0.163	0.381 ± 0.101	0.380 ± 0.232
LC-D	DeBERTa–BiGRU–CRF	1886	0.258 ± 0.250	0.309 ± 0.102	0.373 ± 0.271
HC-C	WCP	1877	0.166 ± 0.373	0.451 ± 0.185	0.171 ± 0.231
HC-C	DeBERTa–BiGRU–CRF	1877	0.005 ± 0.152	0.336 ± 0.089	0.341 ± 0.327
HC-D	WCP	1637	0.051 ± 0.198	0.355 ± 0.099	0.412 ± 0.238
HC-D	DeBERTa–BiGRU–CRF	1637	0.238 ± 0.183	0.303 ± 0.101	0.464 ± 0.232
Overall	WCP	7362	0.110 ± 0.297	0.405 ± 0.162	0.277 ± 0.258
Overall	DeBERTa–BiGRU–CRF	7362	0.106 ± 0.227	0.335 ± 0.131	0.287 ± 0.299

Table 9: Boundary-prediction diagnostics for the two fine-tuned detectors on the common content-filling test set. Here, n denotes the number of evaluated documents. No predicted boundary is the percentage of documents with zero predicted boundaries; Mean predicted and Mean gold are the document-level average numbers of predicted and gold boundaries, respectively. Mean count error is defined as gold minus predicted boundaries, so negative and positive values indicate over-segmentation and missed boundaries, respectively.

Condition	Detector	n	No predicted boundary (%)	Mean predicted	Mean gold	Mean count error
LC-C	WCP	1962	4.18	4.497	1.000	-3.497
LC-C	DeBERTa–BiGRU–CRF	1962	70.90	0.629	1.000	0.371
LC-D	WCP	1886	5.51	4.073	4.914	0.841
LC-D	DeBERTa–BiGRU–CRF	1886	24.23	1.694	4.914	3.221
HC-C	WCP	1877	3.94	4.384	1.000	-3.384
HC-C	DeBERTa–BiGRU–CRF	1877	31.43	1.477	1.000	-0.477
HC-D	WCP	1637	6.66	3.664	5.561	1.897
HC-D	DeBERTa–BiGRU–CRF	1637	2.69	2.561	5.561	2.999
Overall	WCP	7362	5.01	4.174	3.017	-1.158
Overall	DeBERTa–BiGRU–CRF	7362	33.71	1.548	3.017	1.469

D.3 Supplementary Results for Boundary Localization

Comparable overall scores conceal opposite boundary-count biases. As shown in Table 8, the two detectors achieve nearly identical overall Sentence κ values, while DeBERTa–BiGRU–CRF obtains a lower WindowDiff and a slightly higher Boundary F1@1. However, the diagnostics in Table 9 reveal substantially different boundary-count errors: WCP tends to over-segment, whereas DeBERTa–BiGRU–CRF predicts fewer boundaries and produces no predicted boundary for 33.71% of documents. Thus, similar aggregate scores correspond to different localization behaviors.

Collaboration structure changes the dominant localization error. Table 9 shows that the C conditions contain approximately one gold boundary per document. Under these conditions, WCP produces more than four predicted boundaries per document, indicating strong over-segmentation. In the denser D conditions, its mean count error becomes positive, suggesting that the detector does not fully recover the increased number of author transitions. DeBERTa–BiGRU–CRF exhibits a more conservative pattern: it predicts no boundary for 70.90% of LC-C documents, but this proportion decreases to 2.69% under HC-D. Nevertheless, its predicted boundary count remains below the gold count in the D conditions.

Boundary localization requires joint interpretation of quality and count consistency. The results in Tables 8 and 9 show why a single metric is insufficient. DeBERTa–BiGRU–CRF achives a lower overall WindowDiff and a slightly higher Boundary F1@1 in Table 8, but its positive overall count error in Table 9 indicates systematic under-segmentation. Conversely, WCP's higher predicted boundary count reveals over-segmentation that is not fully reflected by its aggregate F1 score. These results indicate that boundary localization should be evaluated jointly in terms of boundary matching and preservation of the overall author-transition structure.

D.4 Supplementary Cross-Condition Generalization Results

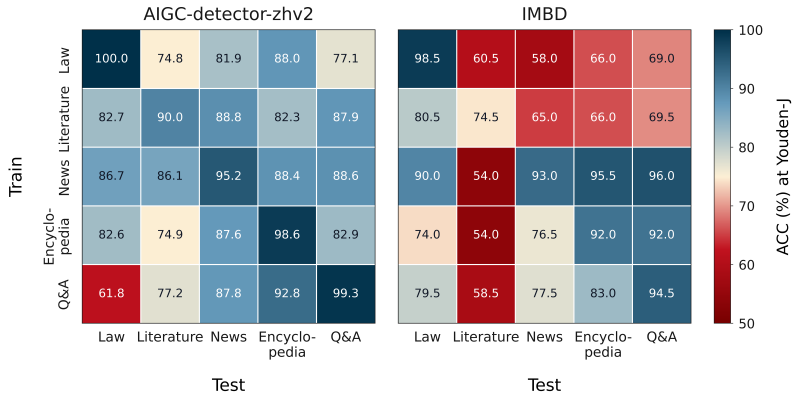

Figure 9: Cross-domain accuracy (%) of AIGC-detector-zhv2 and IMBD with cellwise Youden-optimal thresholds. Rows and columns denote training and test domains, respectively.

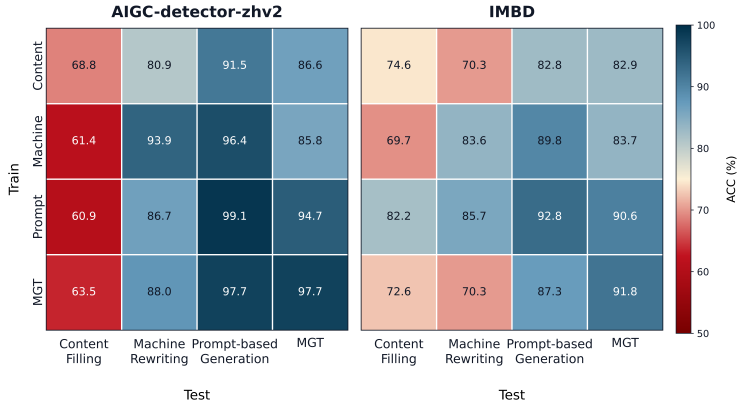

Figure 10: Accuracy (%) across collaboration types for AIGC-detector-zhv2 and IMBD with cellwise Youden-optimal thresholds. Rows and columns denote training and test collaboration types, respectively.

Domain transfer is asymmetric across source and target domains. Figure 9 shows that News provides the strongest cross-domain training source for both detectors, whereas Literature is the most

difficult target, especially for IMBD. This source–target asymmetry suggests that transfer depends on the compatibility between the cues learned from the source domain and the distribution of the target domain, rather than on in-domain accuracy alone.

Content Filling is a detector-specific transfer bottleneck. In Figure 10, Content Filling is the weakest target for AIGC-detector-zhv2, while IMBD retains stronger transfer to this target when trained on Prompt-based Generation. The contrast indicates that the difficulty of detecting locally inserted machine content is not determined solely by the production mode, but also by the cues learned by each detector.

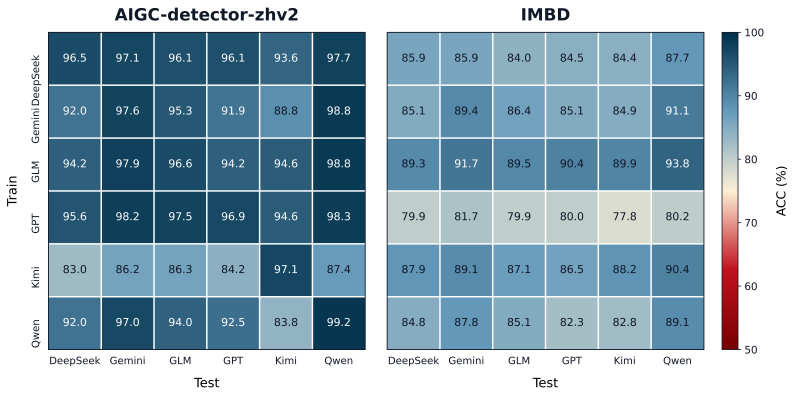

Figure 11: Cross-generator accuracy (%) of AIGC-detector-zhv2 and IMBD with cellwise Youden-optimal thresholds. Rows and columns denote training and test generators, respectively.

Transfer utility and target difficulty are not equivalent. For AIGC-detector-zhv2, training on Content Filling transfers well to Prompt-based Generation and MGT, whereas the reverse directions perform poorly on Content Filling. IMBD exhibits a weaker version of this asymmetry. Thus, a condition may be difficult to recognize as an unseen target while still providing useful training signals for other conditions, making train–test mode matching insufficient for predicting transfer quality.

Cross-generator transfer is governed by detector–source interactions. Figure 11 shows high in-generator performance for both detectors, but off-diagonal transfer varies substantially with the training generator. GPT is the strongest source for AIGC-detector-zhv2 but the weakest source for IMBD, while Kimi causes a marked source-side drop for AIGC-detector-zhv2 without a comparable effect on IMBD. These results show that matched generators do not guarantee the best transfer, and that the transfer value of a generator cannot be assessed independently of the detector.

E Representative Data Construction Examples

We present one human-written news article and its eleven paired variants from C-HAT-240K, all sharing *source_id 0001*. The variants cover MGT, four Content Filling conditions, two Machine Rewriting levels, and four Prompt-based Generation levels.

Paired Data Example from the News Domain

Human-written Source

Original human text

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛8场之后，所有人最关心的一个问题就是帕切科到底如何指挥国安，尤其是在不能依靠对讲机、手机等现代通讯的情况下，帕切科如何指挥成了一个大问题。就在7月2日与陕西的比赛之中，更让人意料不到的是，帕切科竟然在比赛开始之前“消失”了，那么帕切科到底是如何指挥队员们完成这场最艰苦的比赛呢？比赛之前的联席会上，国安队领队魏克兴被明确告之，替补席上的教练员不得使用对讲机与外界进行沟通，因为这个规定是之前早就做出的规定，并不是针对帕切科的停赛，所有球队的替补席人员都不能使用对讲机。这样一来，就比较麻烦了，国安带到西安的对讲机也用不上。

虽然足协没有对使用手机做出限制，但是考虑到比赛现场的信号不好，而且现场特别吵杂，手机通话效果无法保证，最终教练组决定用两个办法传递帕切科的指挥：第一，帕切科把自己的想法写在小纸条，然后传递给教练组；还有就是通过发手机短信的形式发到路易斯的手机上，但是考虑到怕信号不好，手机短信会有延迟，还是决定主要以小纸条这种“鸡毛信”的方式为第一方案，发短信为第二方案。确定了这两个方案之后，帕切科坐在哪里又成为一个大问题。如果是在工体，帕切科可以坐在主席台或者贵宾席上，但在客场，尤其是是西安这个客场，帕切科如果坐在看台上、甚至主席台上都有可能会受到西安球迷的干扰。为了能够让帕切科有一个安静的环境，国安与陕西方面进行了反复的沟通，最终把帕切科安排在了体育场最高处平时几个用来转播的小房间其中之一，而这个小房间也是现场转播的死角，因此在这场比赛之中，不仅现场看不到帕切科坐在哪里，电视镜头也无法捕捉到帕切科的身影，老帕就这样神秘“消失”了。在比赛之中，国安专门准备了两个工作人员传递鸡毛信，其中一个人陪在帕切科身边，在帕切科写完小纸条之后，立刻做电梯来到休息室，再由休息室外的副领队康玉明把小纸条传递给替补席的路易斯。除此之外，帕切科分别对场每一个位置的对应换人进行了交待，这也是在张辛昕受伤情况不明之时，替补席立刻招来小将姜涛准备替补的原因。比赛结束之后，帕切科也承认，自己不能在替补席指挥比赛肯定会给球队造成一定影响，但这种经历自己过去也有过，自己相信任何一个职业教练在执教生涯之中都会遇到这种现象，所以自己也会有相应的办法应对，今天路易斯和其他中方教练代替自己在替补席完成了指挥工作，自己很感谢他们。(宋词发自西安)

MGT

Prefix continuation

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛之后，国安教练组如何在比赛中完成临场指挥，便成了外界关注的焦点。按照足协纪律规定，停赛教练不得进入替补席和更衣室，也不能通过电子通讯设备与场内人员保持联系。也就是说，此前被一些球队采用的对讲机、手机短信等方式都被明确禁止，帕切科即便坐在看台上，也只能像普通观众一样观看比赛。在昨天下午的训练中，国安方面已经开始演练“无主帅指挥”的应急方案。助理教练将在替补席上负责临场调度，而帕切科赛前准备的战术意图，则会通过书面材料提前交给教练组。比赛过程中如果需要调整，教练席只能根据赛前预案和场上形势自行判断。俱乐部人士表示：“规则怎么要求，我们就怎么执行，肯定不会冒险违规。”不过，帕切科一向以激情和临场调整见长，不能在场边大声指挥，对国安显然会有影响。尤其是球队在比赛僵持阶段，换人时机、阵型变化都需要果断决定。为了尽量减少损失，教练组这两天反复研究对手录像，并把可能出现的几种局面列成方案，细化到领先、落后、少打一人等不同情况。队员们也被要求在场上提高自主判断能力，几名老队员将承担更多提醒和组织责任。按照安排，帕切科赛前仍可以参加最后一次战术会议，但在比赛开始前必须离开限制区域。届时他很可能出现在看台包厢内观看比赛。由于不能使用对讲机传达指令，如果确有需要，工作人员也只能在不违反规定的前提下，通过中场休息时的书面意见进行沟通。有人开玩笑说，国安这次恐怕要重新回到“纸条时代”。事实上，足协近来对停赛教练的管理比以往更加严格，目的就是防止处罚流于形式。此前有教练被罚上看台后仍通过耳机指挥，曾引发争议。相关人士强调，停赛的意义就在于剥夺其现场执教权，如果仍能实时控制替补席，处罚就失去了效果。因此本轮比赛监督也会特别留意国安替补席的通讯情况。对于即将到来的比赛，国安队内的态度比较平静。一名队员表示，主教练不能站在场边当然会不习惯，但球队这一周训练安排得很细，大家清楚自己该怎么踢。“帕切科的要求已经讲了很多遍，关键还是我们在场上执行。”俱乐部也希望借此机会证明，球队并不只依赖主帅的临场吼叫，而是有一套相对成熟的战术体系。帕切科本人则没有公开抱怨处罚。熟悉他的人透露，葡萄牙人最担心的不是自己不能指挥，而是球员因此分心。他在训练后特意叮嘱队员，把注意力放在比赛本身，不要被场外话题影响。至于那些写满战术要点的纸条，能否在关键时刻发挥作用，只能等比赛开始后由结果来回答。

Content Filling

LC_D, dispersed filling

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛8场之后，所有人最关心的一个问题就是帕切科到底如何指挥国安，尤其是在不能依靠对讲机、手机等现代通讯的情况下，帕切科如何指挥成了一个大问题。就在7月2日与陕西的比赛之中，更让人意料不到的是，帕切科竟然在比赛开始之前“消失”了，那么帕切科到底是如何指挥队员们完成这场最艰苦的比赛呢？比赛之前的联席会上，国安队领队魏克兴被明确告之，替补席上的教练员不得使用对讲机与外界进行沟通，这也就意味着帕切科即便不坐在替补席上，也不能像以往那样通过对讲机把指令直接传到场边。至于能否打手机，赛前并没有得到明确许可，教练组也不敢冒险，而且现场特别吵杂，手机通话效果无法保证，最终教练组决定用两个办法传递帕切科的指挥：第一，帕切科把自己的想法写在小纸条，然后传递给教练组；还有就是通过发手机短信的形式发到路易斯的手机上，但是考虑到怕信号不好，手机短信会有延迟，还是决定主要以小纸条这种“鸡毛信”的方式为第一方案，发短信为第二方案。确定了这两个方案之后，帕切科坐在哪里又成为一个大问题。如果是在工体，帕切科可以坐在主席台或者贵宾席上，但在客场，尤其是是西安这个客场，帕切科如果坐在看台上、甚至主席台上都有可能会受到西安球迷的干扰。为了能够让帕切科有一个安静的环境，国安与陕西方面进行了反复的沟通，最终把帕切科安排在了体育场最高处平时几个用来转播的小房间其中之一，而这个小房间也是现场转播的死角，因此在这场比赛之中，不仅

现场看不到帕切科坐在哪里，电视镜头也无法捕捉到帕切科的身影，不过在这个位置上，帕切科可以俯瞰全场，国安也专门安排工作人员在房间与替补席之间往返，他把需要提醒的站位、换人和攻防细节写在纸上后，马上由工作人员送到场边，再由路易斯和中方教练向队员传达。由于视野开阔，场上每一个细节都能被他及时发现，这也是在张辛昕受伤情况不明之时，替补席立刻招来小将姜涛准备替补的原因。比赛结束之后，帕切科也承认，自己不能在替补席指挥比赛肯定会给球队造成一定影响，但这种经历自己过去也有过，自己相信任何一个职业教练在执教生涯之中都会遇到这种现象，所以自己也会有相应的办法应对，今天路易斯和其他中方教练代替自己在替补席完成了指挥工作，自己很感谢他们。(宋词发自西安)

LC_C, continuous filling

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛8场之后，所有人最关心的一个问题就是帕切科到底如何指挥国安，尤其是在不能依靠对讲机、手机等现代通讯的情况下，帕切科如何指挥成了一个大问题。就在7月2日与陕西的比赛之中，更让人意料不到的是，帕切科竟然在比赛开始之前“消失”了，那么帕切科到底是如何指挥队员们完成这场最艰苦的比赛呢？比赛之前的联席会上，国安队领队魏克兴被明确告之，替补席上的教练员不得使用对讲机与外界进行沟通，因为这个规定是之前早就做出的规定，并不是针对帕切科的停赛，所有球队的替补席人员都不能使用对讲机。这样一来，就比较麻烦了，国安带到西安的对讲机也用不上。虽然足协没有对使用手机做出限制，但是考虑到比赛现场的信号不好，而且现场特别吵杂，手机通话效果无法保证，最终教练组决定用两个办法传递帕切科的指挥：第一，帕切科把自己的想法写在小纸条，然后传递给教练组；还有就是通过发手机短信的形式发到路易斯的手机上，但是考虑到怕信号不好，手机短信会有延迟，还是决定主要以小纸条这种“鸡毛信”的方式为第一方案，发短信为第二方案。确定了这两个方案之后，帕切科坐在哪里又成为一个大问题。如果是在工体，帕切科可以坐在主席台或者贵宾席上，但在客场，尤其是是西安这个客场，帕切科如果坐在看台上、甚至主席台上都有可能会受到西安球迷的干扰。为了能够让帕切科有一个安静的环境，国安与陕西方面进行了反复的沟通，最终把帕切科安排在了体育场最高处平时几个用来转播的小房间其中之一，而这个小房间也是现场转播的死角，因此在这场比赛之中，不仅现场看不到帕切科坐在哪里，电视镜头也无法捕捉到帕切科的身影，老帕就这样神秘“消失”了。比赛开始之后，帕切科就在这个小房间里通过电视转播和现场视角观察比赛，一旦有新的要求，便立即写在纸条上，由身边的工作人员送到看台通道，再交给守候在附近的国安工作人员，最后传到替补席的教练组手中。由于小房间距离替补席并不算近，每一次传递都需要一定时间，所以帕切科的指令也不可能像平时那样随时、迅速地传达。相比之下，中场休息时的沟通就显得更为重要，教练组把上半场出现的问题集中反馈给他，帕切科再把下半场的调整意见交代清楚。就这样，国安在没有主教练现场坐镇、也无法使用对讲机的情况下，只能依靠这种最原始的办法完成临场指挥。

HC_D, dispersed filling

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛8场之后，所有人最关心的一个问题就是帕切科到底如何指挥国安，国安教练组最初想到的办法，是让老帕坐在看台上通过对讲机随时把战术意图传到替补席，这也是很多人认为最直接、最有效的方式。为了确认这一做法是否可行，俱乐部赛前也专门向比赛监督进行了询问，但得到的答复是否定的：停赛教练不能以任何形式在场边进行指挥，同时按照中超联赛有关规定，所有球队的替补席人员都不能使用对讲机。这样一来，就比较麻烦了，国安带到西安的对讲机也用不上。虽然足协没有对使用手机做出限制，但是考虑到比赛现场的信号不好，而且现场特别吵杂，手机通话效果无法保证，最终教练组决定用两个办法传递帕切科的指挥：第一，帕切科把自己的想法写在小纸条，然后传递给教练组；还有就是通过发手机短信的形式发到路易斯的手机上，但是考虑到怕信号不好，所以最可靠的办法还是“纸条传令”。不过，用纸条也必须解决一个问题：帕切科到底坐在什么位置，既能够看清场上情况，又能保证纸条尽快送到替补席。赛前，国安方面为此费了不少周折，为了能够让帕切科有一个安静的环境，国安与陕西方面进行了反复的沟通，最终把帕切科安排在了体育场最高处平时几个用来转播的小房间其中之一，而这个小房间也是现场转播的死角，因此在这场比赛之中，不仅现场看不到帕切科坐在哪里，电视镜头也无法捕捉到帕切科的身影，老帕就这样神秘“消失”了。在比赛之中，国安专门准备了两个工作人员传递鸡毛信，其中一个人陪在帕切科身边，在帕切科写完小纸条之后，立刻做电梯来到休息室，再把纸条交给等候在那里的另一名工作人员，由他迅速送到替补席交给路易斯。由于从看台最高处到替补席路程不短，这种传递方式难免有些滞后，工作人员还要避开球迷和现场安保通道，因此老帕在上半时很少频繁调整，更多是在中场休息时把自己的意见集中告诉教练组。不过从比赛过程看，替补席对他的意图执行得比较坚决。赛后谈到这种特殊的指挥方式，帕切科表示，今天路易斯和其他中方教练代替自己在替补席完成了指挥工作，自己很感谢他们。(宋词发自西安)

HC_C, continuous filling

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯当帕切科被停赛8场之后，所有人最关心的一个问题就是帕切科到底如何指挥国安，尤其是在不能依靠对讲机、手机等现代通讯的情况下，帕切科如何指挥成了一个大问题。就在7月2日与陕西的比赛之中，更让人意料不到的是，帕切科竟然在比赛开始之前“消失”了，那么帕切科到底是如何指挥队员们完成这场最艰苦的比赛呢？比赛之前的联席会上，国安队领队魏克兴被明确告之，替补席上的教练员不得使用对讲机与外界进行沟通，因为这个规定是之前早就做出的规定，并不是针对帕切科的停赛，所有球队

的替补席人员都不能使用对讲机。这样一来，就比较麻烦了，国安带到西安的对讲机也用不上。按照足协的处罚规定，帕切科不能进入替补席，也不能在赛前、赛中进入休息室进行布置，因此国安方面只能提前把各种可能出现的情况都做成预案。赛前踩场和准备会，助理教练已经把首发、换人以及定位球安排传达给队员，真正到了比赛中，如果临场出现变化，就只能靠看台上的观察和替补席之间最原始的方式联系。开赛前，记者在贵宾席和看台上都没有看到帕切科的身影，甚至连国安工作人员也一度表示“不知道他在哪里”。原来，为了避免引起不必要的争议，帕切科并没有像外界想象的那样坐在醒目的位置遥控指挥，而是选择在一个相对隐蔽的地方观看比赛。由于不能使用对讲机，他只能把自己的意见写在纸上，再由工作人员辗转送到替补席。这样的传递显然无法做到及时，往往等纸条送到场边时，场上的形势已经发生了变化。比赛中，替补席上的助理教练更多是按照赛前既定部署来调整。上半场国安被对手压制时，帕切科显得非常着急，但他能做的也只是把对防守站位和中场逼抢的提醒写下来，让人送下去。到了下半场，国安需要换人加强进攻，相关意见同样是通过纸条完成。相比过去帕切科在场边大声吼叫、不断用手势提醒队员，这种方式无疑让他的作用被大大削弱。不过，国安队员赛后表示，虽然主教练不在身边，但大家赛前已经清楚他的要求，“他平时训练里强调得很多，所以比赛中还是按照他的思路去踢。”对于接下来多场比赛仍将面临同样局面，国安俱乐部也只能继续寻找合规的办法，在不违反纪律的前提下尽量保证教练组与球队之间的沟通。

Machine Rewriting

Light Rewriting

帕切科用对讲机指挥球队被禁传达战术只能靠纸条新浪体育讯帕切科被停赛8场之后，所有人最关心的问题就是他将如何指挥国安。尤其是在不能依靠对讲机、手机等现代通讯工具的情况下，帕切科如何进行临场指挥成了一个大问题。7月2日与陕西的比赛中，更让人意想不到的是，帕切科竟然在比赛开始前“消失”了。那么，他到底是如何指挥队员们完成这场艰苦比赛的呢？比赛之前的联席会上，国安队领队魏克兴被明确告知，替补席上的教练员不得使用对讲机与外界沟通。因为这一规定早已做出，并不是针对帕切科的停赛，所有球队的替补席人员都不能使用对讲机。这样一来，事情就变得比较麻烦，国安带到西安的对讲机也用不上了。虽然足协没有对使用手机做出限制，但考虑到比赛现场信号不好，而且现场特别嘈杂，手机通话效果无法保证，最终教练组决定用两个办法传递帕切科的指挥：第一，帕切科把自己的想法写在小纸条上，再传递给教练组；第二，通过手机短信的形式发到路易斯的手机上。但考虑到信号可能不好，短信会有延迟，教练组还是决定以小纸条这种“鸡毛信”的方式作为第一方案，发短信作为第二方案。确定这两个方案之后，帕切科坐在哪里又成了一个大问题。如果是在工体，帕切科可以坐在主席台或者贵宾席上，但在客场，尤其是在西安这个客场，帕切科如果坐在看台上，甚至是主席台上，都有可能受到西安球迷的干扰。为了让帕切科有一个安静的环境，国安与陕西方面进行了反复沟通，最终把帕切科安排在体育场最高处几个平时用来转播的小房间之一。这个小房间也是现场转播的死角，因此在这场比赛中，不仅现场看不到帕切科坐在哪里，电视镜头也无法捕捉到他的身影，老帕就这样神秘“消失”了。比赛中，国安专门准备了两名工作人员传递鸡毛信，其中一人陪在帕切科身边。帕切科写完小纸条后，工作人员立刻坐电梯来到休息室，再由休息室外的副领队康玉明把小纸条传递给替补席上的路易斯。除此之外，帕切科还分别对场上每一个位置的对应换人进行了交代，这也是在张辛昕受伤情况不明时，替补席立刻招来小将姜涛准备替补的原因。比赛结束之后，帕切科也承认，自己不能在替补席指挥比赛，肯定会给球队造成一定影响。但这种经历自己过去也有过，他相信任何一名职业教练在执教生涯中都会遇到这种情况，所以自己也会有相应的办法应对。今天，路易斯和其他中方教练代替自己在替补席完成了指挥工作，自己很感谢他们。(宋词发自西安)

Deep Rewriting

帕切科遭禁赛无法用对讲机指挥国安战术传达只能靠纸条新浪体育讯帕切科被停赛8场后，外界最关注的问题之一便是：无法坐上替补席的他，将如何指挥北京国安？尤其是在不能使用对讲机、手机等现代通讯手段直接沟通的情况下，帕切科的临场指挥方式更成了焦点。7月2日客场对阵陕西的比赛中，更令人意外的是，帕切科在赛前仿佛“消失”了一般。那么，他究竟是怎样带领球队完成这场艰苦比赛的呢？赛前联席会上，国安队领队魏克兴被明确告知，替补席上的教练员不得使用对讲机与外界沟通。事实上，这项规定早已有之，并非专门针对帕切科的停赛处罚，所有球队的替补席人员都不能使用对讲机。如此一来，国安此行带到西安的对讲机便无法派上用场，临场指挥的难度也随之增加。虽然足协并未限制使用手机，但考虑到比赛现场信号状况难以保证，加之看台环境嘈杂，通话效果并不稳定，国安教练组最终决定采用两套方案来传递帕切科的指令。第一种方式是由帕切科将自己的想法写在小纸条上，再传递给教练组；第二种方式则是通过手机短信发到助理教练路易斯的手机上。不过，由于担心短信可能因信号问题出现延迟，教练组最终将“小纸条”这种类似“鸡毛信”的方式确定为首选方案，短信则作为备用手段。确定传递方式后，帕切科坐在哪里又成了新的难题。如果是在工体，他可以坐在主席台或贵宾席上观察比赛；但在客场，尤其是在西安这样的环境中，帕切科若坐在看台甚至主席台上，都可能受到主队球迷的干扰。为了给他提供一个相对安静的指挥环境，国安与陕西方面进行了多次沟通，最终将帕切科安排在体育场最高处、平时用于转播的几个小房间之一。这个位置恰好也是现场转播的死角，因此整场比赛中，不仅现场观众看不到帕切科在哪里，电视转播镜头也无法捕捉到他的身影，老帕就这样神秘地“消失”了。比赛期间，国安专门安排了两名工作人员负责传递“鸡毛信”。

其中一人始终陪在帕切科身边，等他写好纸条后，立刻乘电梯将指令送到休息室，再由等候在休息室外的副领队康玉明转交给替补席上的路易斯。除此之外，帕切科赛前还对场上各个位置可能出现的换人调整作了详细交代。这也解释了为何在张辛昕受伤情况尚不明确时，替补席能迅速叫来小将姜涛准备替补登场。赛后，帕切科承认，自己无法在替补席现场指挥，肯定会对球队造成一定影响。不过他也表示，类似经历自己过去曾经遇到过，相信任何一名职业教练在执教生涯中都可能面对这样的情况，因此他也有相应的应对办法。帕切科表示，本场比赛路易斯和其他中方教练代替自己在替补席完成了指挥工作，他对此非常感谢。（宋词发自西安）

Prompt-based Generation

Level 1

由于在停赛期间使用对讲机指挥比赛，帕切科再一次成为外界关注的焦点。昨天，中国足协有关人士明确表示，按照相关纪律规定，处于停赛期的教练员不得以任何方式进入替补席、技术区域，也不得通过通讯设备向场内传递指令。也就是说，在接下来的停赛场次中，帕切科如果想把自己的战术意图传达给球队，只能通过赛前布置以及比赛中由工作人员递送纸条等“原始方式”完成。在上一轮联赛中，因累积处罚无法坐上替补席的帕切科出现在看台包厢内。比赛进行过程中，电视镜头多次捕捉到他手持对讲机与场边人员交流。尽管俱乐部方面解释称，帕切科只是与教练组保持联系，并未直接干预比赛判罚或扰乱秩序，但这一做法仍引发争议。按照足协纪律准则，停赛并不只是“不能坐替补席”，更意味着教练员在比赛期间不得以现场指挥的形式继续履行主教练职责。足协相关部门在赛后对录像和比赛监督报告进行了核查，并向俱乐部进行了口头提醒。有关人士表示，停赛处罚的目的在于限制当事人在比赛中的指挥权，如果通过对讲机、电话等方式继续实时下达指令，实际上等同于规避处罚。“停赛教练可以观看比赛，但不能借助通讯工具指挥球队，这是原则问题。”该人士说。这也让帕切科接下来的临场安排变得更加困难。由于无法使用对讲机，教练组只能在赛前把可能出现的各种情况预先设计好，由临时负责临场指挥的助理教练根据比赛走势作出判断。如果看台上的帕切科发现需要调整，他可以通过工作人员递送纸条，将换人、阵型变化或注意事项写明后送到场边。不过，这种方式不仅效率低，而且存在信息延误，尤其是在比赛节奏较快、局势瞬息万变的情况下，很难做到及时反应。俱乐部方面对此显得比较谨慎。一位队内人士表示，球队会尊重足协规定，之后不会再让帕切科使用任何通讯设备指挥比赛。“主教练停赛确实会给球队带来影响，但规则必须遵守。我们会提前做好准备，让助理教练承担更多责任。”据了解，帕切科也已经在训练中加强了对替补席教练组的分工，重点强调比赛中几种常见局面的应对方案，尽量减少无法现场指挥带来的损失。对于球队而言，帕切科的激情和临场调度一直是重要特点。过去的比赛中，他经常在场边大声提醒队员站位，并根据对手变化迅速作出调整。如今被迫远离替补席，球队能否在缺少主教练实时指挥的情况下保持稳定，将成为下一场比赛的一大看点。球员们则表示，停赛不会改变球队的备战节奏，大家会按照教练组要求执行战术。事实上，教练停赛后通过各种方式“遥控”球队，在国内外赛场都并不鲜见。有的教练选择坐在看台上打电话，有的则安排专人传话，但随着竞赛管理越来越严格，这类做法正受到更明确限制。足协方面强调，今后如果再发现停赛人员使用通讯工具指挥比赛，不排除追加处罚的可能。帕切科能否适应只能靠纸条传达战术的特殊局面，将直接影响球队近期表现。对于他和俱乐部来说，如何在规则框架内完成指挥交接，或许比一场比赛的胜负更需要冷静处理。

Level 2

帕切科停赛后，国安队的临场指挥方式成为外界关注的焦点。在无法像往常一样站在替补席前直接调度球队，也不能使用对讲机与替补席保持联系的情况下，这位葡萄牙主帅依然通过另一种方式参与了比赛指挥：小纸条和手机短信，成为他向场内传递战术意图的重要渠道。由于此前受到停赛处罚，帕切科本场比赛只能坐在看台上观战。按照相关规定，停赛教练不得进入替补席区域，也不能通过对讲机等即时通讯设备与场边工作人员直接交流。对于一支习惯由主教练在场边不断提醒、调整阵型和布置细节的球队来说，这无疑是一次特殊考验。比赛开始后，国安替补席上承担主要指挥职责的是助理教练路易斯以及其他中方教练组成员，他们需要在场边完成换人、战术提醒和情绪管理等工作。从比赛现场情况看，帕切科虽然不在替补席，但并未完全“离开”指挥体系。坐在看台上的他始终密切关注场上局势，不时与身边工作人员交流。每当他认为需要对场上站位、攻防节奏或人员安排进行提醒时，便会将自己的意见写在纸条上，或通过手机短信传递给相关工作人员，再由工作人员转达给替补席。随后，路易斯和中方教练根据场上实际情况，将这些信息转化为具体指令，传递给球员。这种“隔空指挥”并不如场边直接喊话那样迅速，但在特殊规则限制下，已经成为教练组能够采取的有效办法。为了避免信息传递出现延误，国安教练组赛前就进行了预案准备，对比赛中可能出现的不同局面作了分工。路易斯主要负责临场战术执行，中方教练则更多承担与球员沟通、提醒站位以及协助换人的工作。帕切科提供的意见，更多是从整体节奏和战术方向上给予提示。据现场观察，国安替补席在比赛中并未因主教练缺席而显得混乱。路易斯多次走到技术区域内向队员做出手势，提醒中后场保持距离，也在进攻推进受阻时要求边路加快出球速度。中方教练则不断在替补席前与热身球员交流，准备随时根据比赛需要进行人员调整。几次关键换人安排，均由替补席教练组完成执行，而帕切科的意见也通过纸条或短信的方式被及时纳入讨论。对于国安来说，帕切科停赛带来的影响不仅体现在临场指挥方式的变化上，也考验着整个教练团队的协作能力。过去，帕切科在场边以情绪饱满、指挥频繁著称，他的临场反应往往直接影响球队节奏。如今他被迫退居看台，助理教练和中方教练必须承担更多

责任，既要贯彻主教练赛前布置，又要根据场上变化独立作出判断。赛后，教练组内部普遍认为，本场指挥工作总体运行顺畅。虽然沟通方式受到限制，但赛前准备较为充分，替补席与看台之间的信息传递基本保持连贯。路易斯和其他中方教练完成了帕切科缺席情况下的场边指挥任务，也让球队在特殊环境下维持了既定战术框架。帕切科的停赛仍将持续一段时间，国安接下来还可能继续面对类似情况。如何在规则允许范围内保持教练组运转效率，如何让替补席在没有主教练直接坐镇的情况下做出更快速判断，将是球队必须继续解决的问题。至少从目前情况看，小纸条和手机短信虽然不是最理想的指挥方式，却帮助帕切科在看台上延续了对球队的影响，而路易斯和中方教练的执行，也让国安度过了停赛后的首个临场指挥考验。

Level 3

7月2日，在国安客场对阵陕西的比赛中，被停赛8场的帕切科无法坐上替补席指挥，也不能通过对讲机与替补席沟通。为保证临场指令传递，国安教练组赛前制定了“小纸条为主、手机短信为辅”的方案，最终由路易斯和其他中方教练在替补席完成现场指挥。这场比赛的地点在西安。此前，帕切科因受到停赛处罚，无法按照以往方式在替补席直接指挥球队。与此同时，替补席教练员不得使用对讲机与外界沟通的要求，也在赛前被再次明确。国安方面了解到，这一规定并非专门针对帕切科的停赛而设，而是足协此前早已作出的规定，比赛中替补席人员不得通过对讲机同看台或其他区域进行联系。据悉，在比赛前的联席会上，国安方面的魏克兴被告知，替补席上的教练员不能使用对讲机与外界沟通。由于帕切科无法出现在替补席，国安教练组必须寻找其他指挥方式。经过沟通和研究，教练组决定将传递小纸条作为第一方案，将手机短信作为第二方案，以便帕切科在观看比赛后及时把自己的判断和调整意见传递到替补席。为了让帕切科能够尽可能清楚地观察比赛，国安与陕西方面进行了沟通。随后，帕切科被安排在体育场最高处用于转播的小房间之一。这个位置虽然远离替补席，但视野较高，有利于他观察比赛整体走势、队形变化以及各个位置的运行情况。对国安来说，在不能进行直接临场指挥的情况下，选择一个便于判断场上局面的观察点，是保障指挥信息有效传递的前提。比赛过程中，国安按照赛前方案安排工作人员负责传递纸条。帕切科在小房间内观察比赛后，将自己的意见通过身边人员传送到休息室，再由康玉明把信息转交给替补席上的路易斯。路易斯收到指令后，再结合替补席上中方教练的判断，对场上队员作出相应安排。手机短信则作为备用方式，在纸条传递不便或需要补充说明时使用。帕切科赛前已经对场上每个位置可能出现的对应换人作了交待，包括不同位置出现体能、战术或人员调整需要时，替补席应如何应对。张辛昕、姜涛等队员所在位置涉及的调整，也都被纳入预案范围。这样一来，即使帕切科不能在场边直接喊话，替补席仍能根据既定预案和临场传来的信息执行调整。在具体执行中，路易斯成为替补席上接收并落实指令的重要人物。由于帕切科不能坐在教练席，他需要依靠路易斯以及其他中方教练完成现场沟通。魏克兴、康玉明等人则在不同环节中承担协调和传递作用，确保从看台高处到替补席之间的信息链条保持通畅。从比赛组织角度看，国安此次指挥方式是在规则限制下作出的应对。足协关于替补席不得使用对讲机与外界沟通的规定，使球队不能采用最直接的远程联系手段；而停赛处罚又使帕切科无法在替补席亲自指挥。两方面因素叠加，促使国安必须在赛前完成更细致的预案，并在比赛中通过人工传递纸条等方式完成临场沟通。最终，路易斯和其他中方教练在替补席上代替帕切科完成了指挥工作。虽然这种方式无法完全等同于主教练现场坐镇，但国安通过赛前安排、人员分工和信息传递，尽量降低停赛带来的影响。帕切科也承认，自己不能在替补席指挥比赛，肯定会给球队造成一定影响，但他表示自己会有相应办法应对，并感谢路易斯和其他中方教练在比赛中的配合与工作。新浪体育讯，宋词发自西安。

Level 3+Ref.

7月2日，在西安进行的国安与陕西比赛中，因被停赛而无法坐上替补席的帕切科，仍通过事先准备的方式参与球队临场指挥。在替补席教练员不得使用对讲机与外界沟通的规定下，国安教练组最终以小纸条作为第一传递方案、手机短信作为第二方案，将帕切科的指挥信息送到替补席，由路易斯和其他中方教练完成现场执行。帕切科此前被停赛8场，这意味着他不能在比赛中进入替补席指挥国安。围绕停赛后的临场指挥问题，国安在赛前就开始准备相应办法。7月2日赛前联席会上，国安方面被明确告知，替补席上的教练员不得使用对讲机与外界沟通。魏克兴在联席会上获知这一要求后，也将情况反馈给教练组。需要说明的是，替补席人员不得使用对讲机并非针对帕切科本次停赛临时作出的安排，而是此前早已存在的相关规定。在无法通过对讲机直接沟通的情况下，国安教练组很快确定了应对方案。按照安排，帕切科的指挥意见首先通过小纸条传递；如果小纸条传递不便，则以手机短信作为第二选择。这样的安排，是为了尽可能保证主教练对比赛形势的判断能够及时传达到替补席，同时也符合比赛管理规定。为了让帕切科能够更清楚地观察比赛，国安方面赛前还与陕西方面进行了沟通。经过协调，帕切科被安排在体育场最高处、用于电视转播的小房间之一。这个位置可以俯瞰比赛场地，便于他观察双方阵型变化、球员跑位以及各个位置上的临场情况。虽然无法像平时一样在替补席近距离指挥，但这一位置至少为他进行全局判断提供了条件。比赛期间，国安按照赛前确定的流程执行传递工作。俱乐部安排工作人员负责传递小纸条，帕切科身边人员将写有指挥信息的纸条送到休息室，再由康玉明把信息传给替补席上的路易斯。路易斯随后结合替补席现场情况，与其他中方教练一起将相关安排传达到场内球员。整个过程虽然比直接口头指挥更复杂，但国安希望通过这种方式尽量减少帕切科停赛对球队临场运转造成的影响。据悉，帕切科在赛前已对场上每个位置可能出现的对应换人进行了交待。也就是说，在比赛开始前，他已经把不同情况下的人员调整方案告知教练组。这样一来，即

使比赛中无法直接站在场边喊话，替补席上的教练仍可根据预案和帕切科传递来的即时判断作出处理。对于一名被停赛的主教练而言，这种预先部署加临场传递的方式，是国安在现有条件下采取的主要应对办法。在实际指挥中，路易斯和其他中方教练承担了帕切科原本在替补席上的职责。他们不仅要关注场上局势，还要接收和理解来自看台高处的指令，再将这些信息转化为场边指挥。魏克兴、康玉明等教练组成员也都参与到这一沟通链条之中，确保信息从帕切科所在位置传递到替补席，再由替补席落实到比赛当中。帕切科也承认，自己不能在替补席指挥比赛，肯定会给球队造成一定影响。与正常情况下相比，他无法第一时间与球员沟通，也不能直接通过临场动作和喊话调整球队。不过他同时表示，自己会有相应办法应对这一困难，并对路易斯和其他中方教练在比赛中的配合表示感谢。从此次比赛的组织过程看，帕切科停赛后的指挥方式并非单一依赖某一种手段，而是国安教练组在规则限制下作出的系统安排。小纸条和手机短信成为替代对讲机的沟通渠道，高处转播房间成为帕切科观察比赛的位置，路易斯和中方教练则在替补席完成最终执行。相关消息由新浪体育讯报道，宋词发自西安。

Figure 12: A paired example from the News domain, comprising one human-written source and eleven machine-involved variants.