HUMANLY: A Configurable and Traceable Environment for Human-AI Collaborative Writing

Shenzhe Zhu[1,2,*] **Haoqian Zhang**[2,*] **Xu Yang**[1,*] **Jingyu Tang**[1] **Yi Nian**[5]
Xiaoxue Du[4] **Shu Yang**[6] **Alex Pentland**[3,4] **Joachim Baumann**[3] **Jiaxin Pei**[1,3,†]
[1]UT Austin, [2]University of Toronto, [3]Stanford, [4]MIT, [5]USC, [6]KAUST
shenzhe@utexas.edu; jiaxinpei@utexas.edu

Abstract

Teachers, conference chairs, and public readers all judge writing from limited evidence, seeing only a finished document and not the process that produced it. Final text alone cannot reveal whether a document was produced through human typing, AI generation, or mixed human-AI collaboration. Existing process-tracking tools help, but many are tied to host-document histories, provide coarse activity records, and offer limited control over the writing environment. HUMANLY is a writing platform that makes the writing process itself the evidence. Users configure writing environments for personal documents or assigned tasks and draft in a workspace that records writing activity and in-platform AI assistance. HUMANLY can package a completed session into a sealed writing certificate with configuration-aware anomaly behavior review. It can support writing scenarios such as course assignments, peer review, and personal certification. Our user study shows that HUMANLY is helpful across roles, and a red-teaming study shows that the HUMANLY Typing Detector distinguishes human hand typing from automated typing. A live deployment (https://writehumanly.net/) and open-source code (https://github.com/Humanly-Lab/humanly) are available.

1 Introduction

Writing authenticity has become a practical problem in settings where decisions are made from final documents alone. Instructors receive student assignments as documents and need to decide whether students followed course AI-use rules; conference chairs may need to inspect whether reviewers used AI; and public readers may question whether a social media post was written by a person or AI. In each case, the final text is visible, but the process behind it is not. We define writing authenticity as whether the production history of a document matches its authorship and AI-use claims: *who wrote what, with which tools, and under what rules*. This question has become harder as human–AI writing has moved from an edge case to a routine practice. Writers use AI for editing, translation, source understanding, brainstorming, and rewriting. These workflows are not equivalent to asking an AI model to produce the final text.

Writing Process	Why post-hoc detection is insufficient
Human draft + AI polish	Cannot reliably demonstrate how AI is involved.
Human draft + AI translation	Cannot recover the human-authored source before translation.
Human-written AI-style text	Style cues can create false suspicion.
AI draft + human polish	Cannot show that substantive AI generation preceded human editing.

Table 1: Writing process that post-hoc detection cannot reliably identify.

Post-hoc detectors classify final text without observing how it was made, leaving them blind to the writing process. They can misclassify non-native English writing as AI-generated (Liang et al., 2023), make inconsistent errors under translation and obfuscation (Weber-Wulff et al., 2023), and lose reliability under paraphrasing or simple text manipulation (Sadasivan et al., 2025; Perkins et al., 2024). Watermarking is also limited to watermark-enabled generation, not arbitrary mixed human–AI workflows (Kirchenbauer et al., 2023). These findings show why a detector score is too little evidence for policy decisions about mixed writing. A post-hoc detector only estimates whether final text appears AI-generated; it cannot distinguish policy-compliant polishing or translation of a human draft from substantive AI generation followed by human editing. Table 1 summarizes four such histories with different policy implications.

Process-tracking tools provide more relevant evidence than final-text detectors because they record parts of the writing process. Revision-history replay and authorship-report products (Table 2) can

[*] Equal contribution. [†] Corresponding author.

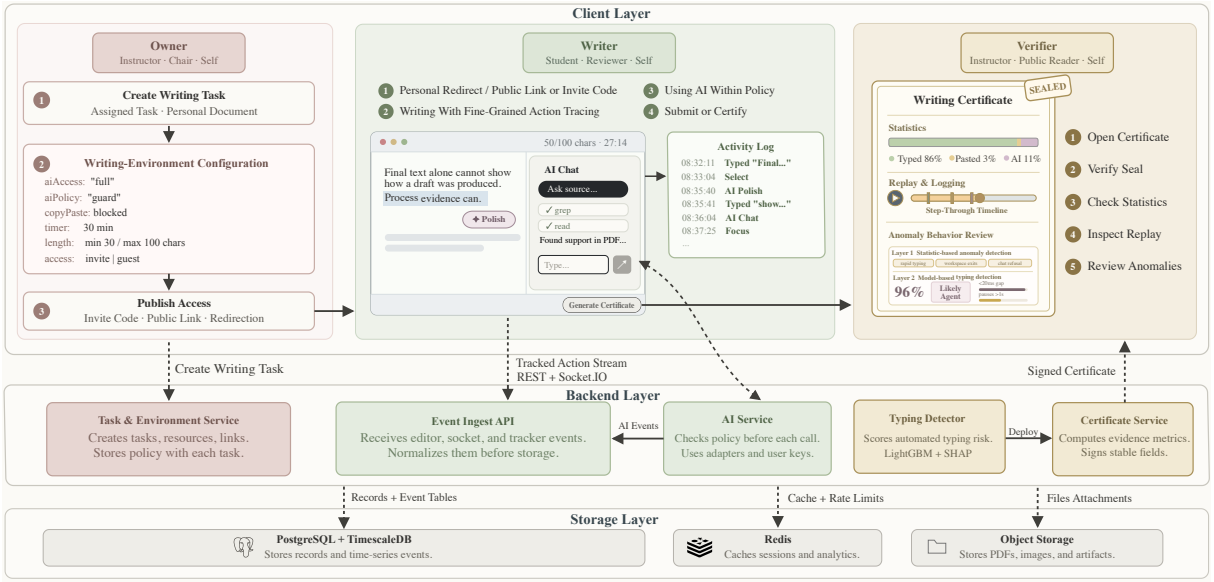


Figure 1: HUMANLY overview. Owners configure a creation mode and writing policy; writers draft under permitted controls while HUMANLY records the session; the backend serves AI responses, coordinates enabled detectors for anomaly behavior review, and issues sealed certificates with logs, replay, and detector results for verifier inspection.

show that text appeared over time, or that some text was typed, pasted, or AI-assisted. However, important practical gaps remain. Many replay-oriented tools are tied to host-document or form environments such as Google Docs, so the process record is often coarse: typed and pasted text may be visible, but workspace activity, detailed AI assistance, and anomaly patterns are often missing or only partially represented. These host environments also limit configurability: the task owner cannot always define the writing environment itself across AI, resource, timing, and length controls. Other systems expose process tracking as a separate product beside AI detection or writing assistance rather than connected in one writing workflow.

We introduce HUMANLY, a writing platform that combines configurable writing environments, fine-grained activity logs, and certificates to make the writing process reviewable end to end (Figure 1). HUMANLY supports three system roles. Owners create personal documents or publish tasks for others; writers draft in the workspace while HUMANLY records writing activity; and verifiers inspect sealed writing certificates as supporting evidence. HUMANLY is designed around four goals. The first is **Flexibility**. HUMANLY should not impose one default writing policy; owners configure writing environments through 14 setting families covering AI policy, resources, budgets, constraints, anomaly behavior review, and access. The second is **Accountability**. HUMANLY makes the writing process inspectable without replacing human judgment. The activity log and authorship statistics show how the final text was composed from typed, pasted, and AI-assisted process components, while anomaly behavior review can combine enabled Anomaly Pattern signals with the HUMANLY Typing Detector for automated-typing risk. A computer-use agent (CUA), for example, can operate software on a user's behalf through graphical or command-line actions.[1] The third is **Verifiability**. Evidence should be issued by the platform rather than asserted by the writer. HUMANLY stores session evidence and environment settings in the certificate. An Ed25519 signature protects selected certificate fields so viewers can detect external modification. The fourth is **Compatibility**. HUMANLY is built for more than one deployment pattern: the same provenance model supports assigned tasks and personal documents, invite-code and public-link distribution, signed-in and guest writing, and MIT-licensed self-deployment.

2 System Architecture and Deployment

HUMANLY is a TypeScript monorepo whose shared schemas connect writing-environment configuration, event ingestion, AI services, certificate generation, detector orchestration, and storage (Figure 1). We call machine-readable backend records *events* and their human-readable log entries *activities*.

Runtime and Storage. First-party clients and the embeddable tracker connect to an Express + Socket.IO backend. Authenticated APIs serve

[1] Examples include OpenAI Codex computer use and Claude Code computer use.

the first-party product workflow, while an open-CORS route ingests external tracker events. PostgreSQL stores product records, TimescaleDB hypertables store high-frequency native editor and tracker events, Redis supports caching and rate limits, and resources use local or cloud storage.

Task and Environment Service. The task service stores the active writing environment with each personal document or assigned task, covering AI, resources, writing constraints, detectors, and access. AI and certificate services resolve this saved record so assistance and later review use the same policy. Route guards enforce ownership, participation, and link or guest eligibility before writes.

Event Ingest and AI Service. Native editor events enter `document_events`, while external tracker events use a batched ingestion route. Before provider dispatch, the AI service resolves the saved environment, checks chat or polish availability, selects the provider/model, and applies the token budget. Responses stream through Socket.IO, while AI sessions, quick-action decisions, and policy refusals are persisted for later review.

Certificate Service and Anomaly Detector. At issuance, the certificate service freezes the session boundary, computes authorship statistics and detector results, and stores the environment snapshot and verification token; logs and replay are served against the same boundary. The service reads the saved detector configuration so disabled detectors remain explicit. Anomaly Pattern derives five statistic-based signals from recorded events and policy conditions. The HUMANLY Typing Detector sends session events to a LightGBM inference service (Ke et al., 2017), which estimates automated-typing probability from keystroke rhythm and editing behavior and returns SHAP-based contributing features (Lundberg and Lee, 2017). Insufficient timing data produces an inconclusive result. Ed25519 signs the protected certificate fields, including detector results when present, and a backend endpoint exposes the public verification key.

Deployment. For local self-hosting, HUMANLY provides a one-line bootstrap command: `curl -fsSL https://writehumanly.net/install.sh | sh` The installer prepares Docker/Compose, source checkout, local secrets, storage, and an admin account, then starts the database, cache, backend, and portals. Production deployments replace local URLs, email, and storage settings.

3 Workflow and Use Cases

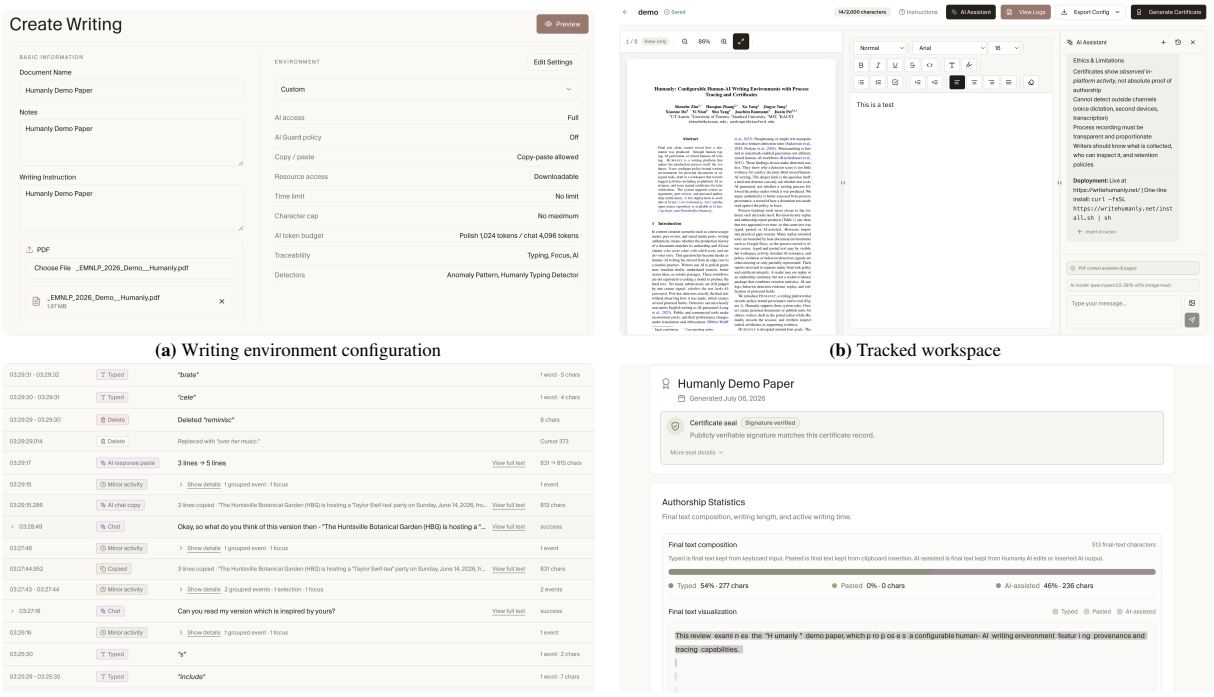
3.1 Workflow Design

Figure 1 organizes the workflow around three roles: Owner, Writer, and Verifier. These are not fixed account types: assigned tasks may distribute them across users, while personal documents may combine the Owner and Writer roles. Figure 2 follows the corresponding interfaces from setup through writing and certificate review.

Owner. The Owner role begins when a user creates a writing environment. Personal-document owners configure a private document in the Writer Portal; assigned-task owners configure a shared task in the Publisher Portal and distribute it through invite codes or public links. Across both paths, owners configure 14 setting families, including resources, constraints, access, four AI modes (off, polish-only, chat-only, and full), and two detectors for anomaly behavior review: Anomaly Pattern and HUMANLY Typing Detector. Table 5 in Appendix C summarizes the setup options. HUMANLY saves this configuration with the session, so the certificate can be interpreted against the rules active during writing.

Writer. The Writer role begins when a user enters a configured workspace, either through their own personal document or through an assigned task shared by another owner. Before drafting, the writer is shown the task instructions and writing rules defined by the owner-set policy. With full mode and source resources enabled, the workspace presents three panels: source PDFs or resources, the editor, and the AI assistant. Writers can use chat to ask questions about the provided resources and apply four AI quick actions to selected text during drafting. The activity log exposes 27 reviewable activity labels across text editing, clipboard activity, workspace status, AI assistance, and anomaly patterns. Table 6 in Appendix C counts these labels by category. When writing is complete, the writer submits the task or generates a certificate from the recorded session.

Verifier. Certificate review begins after issuance. For assigned tasks, issuance follows submission; for personal documents, the writer generates the certificate after finishing the document. A verifier may be the original writer, the task owner, or any public reader with a shared verification link. The certificate evidence includes authorship statistics, environment settings, an activity log, replay, the anomaly behavior review configured for that writing environment, and an Ed25519 integrity seal. Authorship statistics summarize final-text compo-



(a) Writing environment configuration

(b) Tracked workspace

(c) Activity log

(d) Certificate

Figure 2: HUMANLY interfaces: (a) writing environment configuration; (b) tracked workspace in full mode with PDF resources, editor, and AI assistant; (c) activity log; and (d) certificate with authorship statistics and signature.

sition and process input volume; environment settings preserve the policy active during writing; the activity log presents recorded activities; and replay shows how the document changed over time. When both detectors are enabled, anomaly behavior review includes Anomaly Pattern signals and the HUMANLY Typing Detector score; disabled detectors are shown as not enabled. The integrity seal signs selected certificate fields with Ed25519, allowing verifiers to use HUMANLY's public key to check whether those fields were modified after issuance.

3.2 Use Cases

Course assignments. Course assignments follow a familiar learning-management workflow: an instructor publishes a task, students complete it, and the instructor reviews the submission. Standard assignment systems collect the final file, but they rarely show whether students followed AI-use rules while writing. HUMANLY preserves the same assignment flow while adding a configurable writing environment and process evidence that can surface anomaly patterns and automated typing behavior when possible rule violations need inspection.

Peer review. Peer review has a similar publish-write-review flow: an area chair assigns a paper, reviewers write reviews, and chairs inspect the results. AI-assisted reviewing is becoming part of this workflow at several academic conferences (Baumann et al., 2026).[2] Even under such policies, compliance often depends on reviewers following the rules without direct process evidence (Saha et al., 2026). HUMANLY is useful here because chairs can provide an in-platform AI assistant, record how it was used, and inspect process evidence when review quality is questioned.

Social media posting. Social media posts are often judged from final text alone, and recent reporting describes human writers accused of AI use when their prose appears polished or machine-like.[3] In this setting, HUMANLY is useful because a poster can draft as a personal document and share a signed certificate link with readers. Instead of inferring authorship from style alone, readers can inspect the observed writing process behind the post.

4 Comparison with Existing Systems

Table 2 compares HUMANLY with seven related writing-authenticity systems that provide process evidence, authorship reports, or replay for written documents. The rows follow three product areas: a configurable writing environment, a native activity log, and a certificate layer. Ratings indicate whether each feature is supported in public product documentation. Overall, the closest comparator is Turnitin Clarity: it supports instructor-managed

[2] NeurIPS 2026 AI-assisted reviewing experiment; ICML 2026 LLM reviewing policy. [3] WHYY, "How Not to Be Mistaken for a Chatbot"; New York Magazine, "The People Falsely Accused of Using AI".

Area	Feature	HUMANLY	Turnitin	Grammarly	GPTZero	Draftback	Brisk	Integrito	PaperTrail
Writing Environment	Configurable rules	✓	✓						
	Task assignment	✓	✓						
	AI policy	✓	✓						
	Open source	✓							
Activity Log	Text editing	✓	✓	✓	✓	✓	✓	✓	✓
	Clipboard	✓	✓	✓	✓		✓	✓	✓
	Workspace state	✓							
	AI assistance	✓	✓	✓					
	Anomaly pattern	✓	✓		✓			✓	✓
Certificate	Replay	✓	✓	✓	✓	✓	✓	✓	✓
	Authorship statistics	✓	✓	✓	✓	✓	✓	✓	✓
	Typing detector	✓							
	Integrity seal	✓							

Table 2: Feature comparison of writing-authenticity systems by writing environment, activity log, and certificate. Competitor columns refer to the linked product documentation; ✓ means supported, and blank cells mean not found in public documentation. Configurable rules = owner-set environment constraints; AI policy = visible AI-use modes and guard rules; workspace state = focus, selection, or workspace leave/return events; anomaly pattern = statistic-based anomaly behavior pattern derived from logged events and active policy; typing detector = estimate of whether typed input resembles human hand typing or automated typing; integrity seal = Ed25519 certificate signature that can be checked with HUMANLY's public key.

assignments in which students write under AI-use guidance and instructors receive writing-process reports. Grammarly Authorship adds authorship tracking, source attribution, shareable reports, and replay across supported writing surfaces. The other five products rely primarily on Google Docs or browser-extension workflows for writing reports and replay. We therefore score whether each system allows a task owner to configure writing policy before writing, records human and AI activity at sufficient granularity, and packages the result as certificate-style evidence.

Writing Environment. HUMANLY's first difference is that it offers a configurable writing environment rather than depending on a host editor's inherited policies. Owners set resource, timing, length, access, and AI rules before writing. Turnitin Clarity is strongest in this category because it supports instructor-controlled assignment settings and AI-use guidance for student writing. HUMANLY is released for inspection, adaptation, and self-deployment.

Activity Log. HUMANLY's second difference is the scope of the fine-grained activity log. The workspace logs text editing, clipboard, workspace state, AI assistance, and anomaly patterns. Most systems in the table provide text-editing and clipboard records, but workspace state is unique to HUMANLY: it records how the writer uses the workspace, including focus, blur, workspace leave, and return events. Four competitors expose narrower anomaly cues, such as playback flags, writing-pattern evaluation, suspicious-event summaries, or struggle moments.

Certificate. In the certificate layer, HUMANLY links authorship statistics, writing replay, anomaly behavior review, task policy, and the integrity seal. HUMANLY differs in that these fields are bundled into one certificate package rather than split across separate reports or tools. GPTZero illustrates this fragmentation: its Chrome extension offers Google Docs writing logs, an AI assistant, and AI scan, but its AI-composition verification still relies on a prediction scan over final text rather than attribution from the recorded process and in-extension assistant actions.[4] Model-based human typing detection and the integrity seal are unique to HUMANLY in this comparison. The other products do not address whether typed input was produced by human hand typing or automated workspace operation. They also do not protect certificate evidence against later tampering with a public-key signature.

5 Evaluation

We evaluate HUMANLY through a role-based user study and a red-teaming study. The user study tests whether the system is usable across the Owner, Writer, and Verifier roles; the red-teaming study tests whether the HUMANLY Typing Detector separates human hand typing from automated workspace operation in a controlled writing task.

5.1 Role-Based User Study

The role-based user study examines the three roles in Figure 1: Owner, Writer, and Verifier. We recruit three independent groups of Prolific AI Taskers in

[4] In an illustrative manual check of the GPTZero Google Docs workflow (June 2026), text generated through GPTZero's in-extension AI assistant was still rated as human by the separate scan feature.

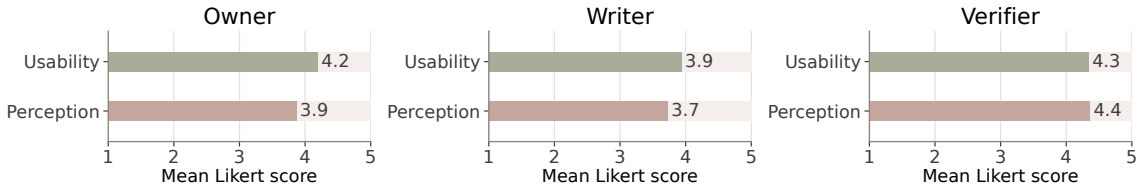


Figure 3: Role-based user study results. Bars report respondent-level mean Likert scores for Usability and Perception. Scale: 1 = Strongly disagree, 2 = Disagree, 3 = Neither agree nor disagree, 4 = Agree, and 5 = Strongly agree.

the Structured Writing category[5]. Each group receives one survey, and quality review leaves 30 valid responses per role. Survey items use a five-point agreement scale from Strongly disagree to Strongly agree. Across roles, **Usability** measures whether participants can use or interpret the relevant interface. **Perception** captures role-specific judgments: Owners rate policy transparency and confidence in rule compliance; Writers rate accountability, authorship proof, and recording anxiety; Verifiers rate fairness and trust in process evidence. Detailed setup appears in Appendix A.

Analysis. Figure 3 shows positive Usability and Perception ratings across all three roles, with every mean above 3 on the five-point agreement scale. Verifiers give the strongest ratings ($M = 4.34$ for Usability and $M = 4.36$ for Perception). This suggests that readers who did not produce the writing can still understand the evidence package. Writer Perception is lower ($M = 3.73$) but still positive. This result reflects the tradeoff in the writer role: participants valued clearer rules and evidence they could later share, but some were also uneasy about detailed recording while they wrote.

5.2 Red-Teaming Study

Motivated by recent work on the security of computer-use agents (CUAs) (Chen et al., 2025; Liao et al., 2025), we perform a red-teaming study to evaluate the HUMANLY Typing Detector. We test one configurable detector in HUMANLY's anomaly behavior review under a policy that requires writers to operate the workspace directly; delegating browser control to a CUA is therefore non-compliant. The detector estimates whether typed interaction resembles human hand typing or automated typing.

Data Collection. We collect human-operated submissions through Prolific. For agent-operated submissions, we use OpenAI Codex's computer-use capability with GPT-5.5 as the base model to control the workspace. Copy-paste remains allowed, and a 20-minute limit standardizes task completion. Detailed settings appear in Appendix B.

[5] Prolific AI Taskers, Structured Writing

Baselines and Metrics. We compare the HUMANLY Typing Detector against three zero-shot LLM-as-judge baselines that score serialized activity logs (Table 3). Human FPR is our primary operating constraint: falsely flagging a compliant writer can create unwarranted suspicion, so the detector must keep this error low while identifying automated typing. We therefore use a conservative threshold and report human false-positive rate (FPR; lower is better), agent true-positive rate (TPR; higher is better), AUROC (higher is better), and inference time (lower is better). More details are in Appendix B.

Method	Human FPR ↓	Agent TPR ↑	AUROC ↑	Inference time (s) ↓
GPT-5.1	**0.0%**	15.0%	0.563	3.4
Claude Sonnet 4.6	**0.0%**	60.0%	0.838	12.7
Claude Opus 4.8	5.9%	**90.0%**	0.965	11.5
Ours (LightGBM)	**0.0%**	85.0%	**0.994**	**0.0024**

Table 3: HUMANLY Typing Detector compared with LLM-as-judge baselines on red-team submissions.

Results. At this operating point, HUMANLY achieves 0.0% human FPR, 85.0% agent TPR, and the best AUROC. Claude Opus 4.8 reaches a higher agent TPR (90.0%) but flags 5.9% of human-operated submissions and has lower AUROC. HUMANLY averages 0.0024 seconds per submission, approximately $4{,}800\times$ faster than Opus 4.8 and at least $1{,}400\times$ faster than every LLM judge in our setup. Local LightGBM inference also avoids a paid model call for each full activity log. The detector stores SHAP-based contributing features, giving verifiers feature-level explanations of which typing behaviors drove the score.

6 Conclusion

HUMANLY reframes writing authenticity as policy-bound process evidence. It combines configurable writing environments, activity logs, anomaly behavior review, replay, and public-key verification. Across course assignments, peer review, and personal certification, the workflow ties owner-defined rules to evidence. Our studies show cross-role usability and reliable separation of human and automated typing with few false alarms.

Ethics and Broader Impact Statement

Broader Impact. HUMANLY shifts writing-authenticity decisions away from final-text suspicion and toward process evidence. This can benefit writers whose polished, translated, or non-native English work is questioned, and it can help instructors, editors, and public readers review mixed human–AI writing with more context. The same capability creates risk: process evidence can become intrusive if recording becomes routine surveillance or certificates become disciplinary shortcuts. HUMANLY should therefore be deployed as a policy-transparent review aid, not as general monitoring or an automatic misconduct judge.

Limitation: What Certificates Can and Cannot Show. A HUMANLY certificate records observed in-platform activity; the certificate is not an authorship verdict. It reports the recorded process, policy context, statistic-based anomaly detection, and, when enabled, a model-based human typing score. The certificate cannot prove intention or rule out every outside channel. Page-visibility records show that the HUMANLY workspace was hidden or visible again; they do not identify which tab, website, application, or device the writer used while away. A writer can transcribe external AI output, dictate it by voice, or route text through another device. Certificates therefore support inspection of observed activity rather than proof that no outside assistance occurred. The red-teaming study evaluates browser or GUI-agent operation of the HUMANLY editor, not every off-platform assistance channel. HUMANLY only records activity inside HUMANLY; tracker-based documents carry less detail than native-editor documents, and writing outside HUMANLY leaves no record.

Recording Governance. Process recording captures typing behavior, clipboard activity, workspace activity, formatting, and AI assistance. Writers should know which activity categories are logged, who can inspect them, how long they are retained, and whether public verification exposes a summary or a full replay. HUMANLY surfaces the rules at document setup and keeps only the permitted controls visible while writing. Deployments still need clear local policies for consent, access, retention, appeals, and reader training, especially in classrooms and peer review. Our studies follow the same principle: recording is disclosed up front, and participants are asked to interpret evidence without treating the certificate as an automatic verdict. Personally identifiable information is limited to participant payment and study administration.

References

Joachim Baumann, Jiaxin Pei, Sanmi Koyejo, and Dirk Hovy. 2026. Stop automating peer review without rigorous evaluation. In *Post-AGI Science and Society Workshop*.

Ada Chen, Yongjiang Wu, Junyuan Zhang, Jingyu Xiao, Shu Yang, Jen-tse Huang, Kun Wang, Wenxuan Wang, and Shuai Wang. 2025. A survey on the safety and security threats of computer-using agents: JARVIS or Ultron? *arXiv preprint arXiv:2505.10924*.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

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

Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. 2023. GPT detectors are biased against non-native English writers. *Patterns*, 4(7):100779.

Zeyi Liao, Jaylen Jones, Linxi Jiang, Yuting Ning, Eric Fosler-Lussier, Yu Su, Zhiqiang Lin, and Huan Sun. 2025. Redteamcua: Realistic adversarial testing of computer-use agents in hybrid web-OS environments. *arXiv preprint arXiv:2505.21936*.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Jiaxin Pei, José Ramón Enríquez, Umar Patel, Alia Braley, Nuole Chen, Lily Tsai, and Alex Pentland. 2025. DELIBERATION.IO: Facilitating democratic and civil engagement at scale with open-source and open-science. Working paper.

Mike Perkins, Jasper Roe, Binh H. Vu, Darius Postma, Don Hickerson, James McGaughran, and Huy Q. Khuat. 2024. Simple techniques to bypass GenAI text detectors: Implications for inclusive education. *International Journal of Educational Technology in Higher Education*, 21:53.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2025. Can AI-generated text be reliably detected? stress testing AI text detectors under various attacks. *Transactions on Machine Learning Research*. Also available as arXiv:2303.11156.

Rounak Saha, Gurusha Juneja, Dayita Chaudhuri, Naveeja Sajeevan, Nihar B Shah, and Danish Pruthi. 2026. Policies permitting LLM use for polishing peer reviews are currently not enforceable. In *Forty-third International Conference on Machine Learning*.

Debora Weber-Wulff, Alla Anohina-Naumeca, Sonja Bjelobaba, Tomáš Foltýnek, Jean Guerrero-Dib, Olumide Popoola, Petr Šigut, and Lorna Waddington. 2023. Testing of detection tools for AI-generated text. *International Journal for Educational Integrity*, 19(1):26.

Shenzhe Zhu, Shu Yang, Michiel A. Bakker, Alex Pentland, and Jiaxin Pei. 2025. Can AI truly represent your voice in deliberations? a comprehensive study of large-scale opinion aggregation with LLMs. *arXiv preprint arXiv:2510.05154*.

A User-Study Survey Administration

We administer three role-specific surveys in Deliberation.io (Pei et al., 2025; Zhu et al., 2025). Prolific[6] routes qualified Structured Writing AI Taskers to the corresponding survey, and participants receive a completion code upon submission. Owners create and publish a task, Writers complete an assigned task in HUMANLY, and Verifiers review fixed certificate screenshots and context. All agreement items use a five-point scale, and we retain 30 valid responses per role. For Owners and Writers, we manually inspect submitted task or certificate links for on-task completion. Because Verifiers review fixed materials, we instead exclude submissions completed in under five minutes (expected: 10–20 minutes). Dimension scores are respondent-level means. "I did not use it" responses to optional Writer AI-usefulness items are excluded item-wise, and the Writer anxiety item is reverse-coded before computing Perception. The full questionnaire is available online.[7]

B Red-Teaming Study Details

Task and Agent Setup. We collect 20 human-operated and 20 agent-operated submissions. After quality filtering, 17 human-operated and 20 agent-operated submissions remain usable for the common evaluation. Red-team sessions use a standardized TOEFL "Writing for an Academic Discussion" prompt in the native HUMANLY editor, with the AI mode set to full, copy-paste allowed, a 20-minute window, and a 1000–2000 character requirement. The test prompt uses a topic disjoint from the topics used to train the HUMANLY Typing Detector. For automated-operation sessions, a computer-use agent (CUA) controls the browser and completes the HUMANLY writing task. We instantiate this condition through the OpenAI Codex platform,[8] using GPT-5.5 as the base model, under two prompting strategies: Direct and Human-like. The prompt text is maintained online.[9]

Baselines and Metrics. The LLM-as-judge baselines score each serialized activity log zero-shot, using GPT-5.1, Claude Sonnet 4.6, and Claude Opus 4.8; Claude Opus 4.8 runs with high reasoning effort. Each judge receives a generic task description with no hand-engineered detection cues or labeled examples. The judge prompt is maintained online.[10] The HUMANLY Typing Detector threshold is chosen on held-out validation submissions to keep the human false-positive rate low, since a false flag directs review toward a compliant writer. Treating agent-operated submissions as positive, Table 4 defines the thresholded outcomes. We report four metrics. **Human FPR** is $FP/(FP+TN)$, the fraction of human-operated submissions wrongly flagged as agent-operated. **Agent TPR** is $TP/(TP+FN)$, the fraction of agent-operated submissions correctly flagged. **AUROC** summarizes threshold-free separation: the probability that a randomly chosen agent-operated submission receives a higher detector score than a randomly chosen human-operated one, where 0.5 is chance and 1.0 is perfect separation. **Inference time** is the mean wall-clock time to score one submission. For the LLM judges this is the per-submission API round-trip through OpenRouter (network plus model generation), whereas for the HUMANLY Typing Detector it is local feature extraction plus a single LightGBM prediction with no network call.

	Pred. agent	Pred. human
Actual agent	TP	FN
Actual human	FP	TN

Table 4: Confusion matrix for thresholded human typing detection.

C Configuration and Capture Details

Table 5 lists the main environment controls exposed by HUMANLY, and Table 6 counts the 27 reviewable activity labels in the native writing editor. The main paper summarizes these fields because the exact configuration differs between personal writing and assigned tasks.

[6] Prolific [7] HUMANLY user-study question list.

[8] OpenAI Codex computer use. [9] HUMANLY red-team Codex prompts. [10] HUMANLY LLM-as-judge prompt.

Setting	Personal	Task	Values
Files/resources	✓	✓	Source PDFs, task PDFs, and writer-provided resources.
Resource access	✓	✓	Downloadable or view-only PDFs; view-only uses short-lived tokens and canvas rendering.
Preset/import	✓	✓	Default, custom, and JSON import/export configurations.
AI access	✓	✓	Off, polish-only, chat-only, or full assistance.
AI guard policy	✓	✓	Optional rejection rules for agent chat in chat-only or full modes.
AI provider/model	✓	✓	User API key; Together, OpenRouter, OpenAI, Anthropic; curated model/capability allowlists.
AI budgets	Tokens	Tokens +requests	Shortcut/chat token budgets; per-user task request cap.
Copy/paste	✓	✓	Copy, cut, and paste allowed or blocked.
Writing timer	✓	✓	Optional countdown; timed task attempts can auto-submit at expiry.
Anomaly behavior review	✓	✓	Enable Anomaly Pattern, HUMANLY Typing Detector, or both.
Length bounds	Max	Min/max	Final character limits for personal writing or assigned tasks.
Task attempts	No	✓	Single durable attempt or restart-allowed mode with a maximum attempt count.
Availability	No	✓	Task start and end window.
Guest link	No	✓	Public link can require sign-in or allow guest submissions.

Table 5: Configurable writing-environment settings. "Task" refers to admin-created assigned tasks and public share links.

Category	Count	Native editor
Text editing	5	Typed text; deletions; replacements; line breaks; blank-line insertions.
Clipboard / formatting	4	Paste; copy; cut; formatting changes.
Workspace status	5	Workspace leave; workspace return; editor focus; editor unfocus; text selections.
AI assistance	8	AI chat; AI quick actions (Fix grammar; Improve writing; Simplify; Make formal); AI inserted; AI chat copy; AI response paste.
Anomaly patterns	5	Rapid text accumulation; untracked text source; frequent workspace exits; blocked copy-paste attempts; chat refusals.

Table 6: Reviewable activity labels in the native writing editor and certificate evidence. The five anomaly patterns are derived from logged events and certificate metrics rather than additional raw event types.