Keystroke Dynamics and NLP: Towards a Hybrid Detector of AI-Generated Text

Matteo van Ypersele
ENSAE Paris 3A
matteo.vanyperseledestriho@ensae.fr

Abstract

Current detectors of AI-generated text rely almost exclusively on the *finished product*. We ask whether the *writing process*, observable via keystroke logs, can provide a complementary signal. Starting from OPENLLMTEXT (60 000 human-written / 60 000 ChatGPT paragraphs) and enriching each paragraph with 20 keystroke features drawn from the KEY LOG corpus, we compare three lightweight classifiers: (i) a RoBERTa-base model fine-tuned on text only, (ii) a 30 000-parameter MLP fed only with the keystroke vector, (iii) a late-fusion network concatenating both representations. On a held-out test set of 10 375 paragraphs, the keystroke-only MLP already achieves an **F_1 of 0.994** and a **ROC-AUC of 0.997** with 3.6 % false positives, outperforming the text-only RoBERTa baseline (0.848 / 0.866, 22 % FP). However, note that transcribers did not attempt to conceal their status. A naïve fusion of both views yields an **F_1 of 0.954** and a **ROC-AUC of 0.989**, but drives the FP-rate up to 28.4 %.

Preprint. Under review.

1 Introduction

The recent rise of *Large Language Models* (LLMs) such as GPT-4 or DeepSeek is profoundly transforming writing practices. In academic settings, students can generate a "ready-made" essay in seconds, undermining take-home assignments. In professional contexts, online exams and writing certifications become equally vulnerable: a candidate can delegate a cover letter or analytical report to a conversational agent without leaving visible traces.

Existing detectors rely almost exclusively on the *finished product*. They measure text perplexity or apply a classifier trained on human/AI text pairs. Yet such methods are easily fooled by heavy paraphrasing, back-and-forth translation, or manual rewrites that obscure the AI's statistical signature.

We argue that a robust detector should account for the *writing process*. Pauses, edits, and typing bursts reflect human cognitive load and are expensive to mimic. To capture this signal, we propose a locked-down web interface that logs every insertion, deletion, and timing event at millisecond precision, then aggregates these events into twenty privacy-respectful metrics (no individual character content is stored).

Our research question is: *Does combining these keystroke metrics with frozen linguistic embeddings improve AI-generated text detection?* To answer this, we (1) enrich the public OPENLLMTEXT corpus with authentic keystroke logs, (2) train a lightweight fusion classifier based on *RoBERTa-base*, and (3) compare its performance to two baselines: text-only and keystroke-only.

1.1 Ready-Made LLM-Based Detectors

Early AI detectors used the *perplexity gap*: a text is scored by two models of different sizes, and a large gap signals likely AI generation [Zellers et al., 2019]. With transformers, two strategies dominate. The first queries a proprietary LLM (GPT-4, Claude-3) in zero-shot mode. Li et al. [2023] report 96–98 % accuracy on student essays, but API dependence, network latency, and lack of explainability hinder operational deployment. The second fine-tunes open models on labeled datasets: Solaiman et al. [2019] trained GPT-2 "small" on 250 k human/GPT-2 pairs and achieved 84 % accuracy on short news.

Three avenues emerge: (i) **Transformer-only classifiers**: Ippolito et al. [2020] adapt RoBERTa-base on synthetic Reddit (Grover) and get 92 % AUC, but only 71 % out-of-domain; on HC3, Zhou et al. [2023] show ELECTRA-large gains four F_1 points via corruption pretraining. (ii) **Hybrid ensembles** combining Transformer probabilities and stylometry: Mikros and Stamatatos [2023] fuse RoBERTa-large and ELECTRA-large outputs with 92 classical indices in a Gradient Boosting, achieving 95 % accuracy and reducing FP rate by 7 points on TOEFL. (iii) **Entropy-gap methods**: Tan and Wang [2023] retrain GPT-NeoX-20B to predict log-ratios of its own perplexity versus a smaller model, getting 97 % AUC in-domain but dropping to 62 % after paraphrase, highlighting finished-product fragility [Mitchell et al., 2023].

Why RoBERTa-base? We sought an open-weights encoder referenced on GLUE but compact enough for real-time inference on a single V100 GPU. Pretrained on 160 GB of tokens, RoBERTa-base (125 M parameters) outperforms BERT-base by three F_1 points [Liu et al., 2019] with the same memory footprint. A light fine-tune competes well with larger models like ELECTRA-large or DeBERTa-XL [Zhou et al., 2023], at roughly half the inference cost. Freezing embeddings and half the layers cuts adaptable weights by 60 %, limits overfitting on our 52 k paragraphs, and reduces training time by a factor of three. Yet even robust encoders remain vulnerable to paraphrase and translation—hence the need for process signals.

1.2 Exploiting Keystroke Logs

Psycholinguistic research has long established that pause durations, typing burst structure, and revision patterns reflect cognitive load [Conijn et al., 2019]. On the Villani corpus (338 copied texts vs. 416 free emails), Conijn and Villani [2019] already differentiate genres with 78 % accuracy using only character-level pauses. A larger scale effort by Crossley et al. [2024] collected 1 000 argumentative essays—500 authentically written by 4 992 MTurk workers and 500 transcribed word-for-word by another group. Logged at 1 ms precision in a walled garden, the keystrokes are analyzed with INPUTLOG, yielding 155 measures covering writing rate, short and long pauses, bursts,

insertions/deletions, and temporal variability. After removing missing-value and highly collinear features, 65 z-scored indicators remain. A 500-tree random forest achieves 99 % accuracy and identifies 20 key features (short/long pauses, product/process ratio, insertion/deletion volumes, typing rate variance) that explain 93 % of permutation importance—demonstrating that process signals alone are highly discriminative and robust to paraphrase.

1.3 Fusion of Finished Product and Process: An Open Field

Few studies directly merge text surface and typing dynamics. Li and Sun [2022] concatenated BERT embeddings with five pause metrics in a small MLP; on 8 000 internal emails, the procedural signal only added two F_1 points without cross-domain gains. To enhance robustness, Hems and Lex [2023] propose a late fusion between DetectGPT zero-shot and a keystroke random forest, reducing false positives by threefold on exam essays, though relying on GPT-4 API. Simultaneously, Lopes et al. [2023] show that combining eye-tracking and keystrokes halves cognitive load prediction error, suggesting strong complementarity.

Despite these advances, key questions remain: does the gain hold on a massively out-of-domain, multi-author corpus? Is simple concatenation followed by a linear head sufficient, or is cross-modal attention needed? And do the 20 features generalize to newer LLMs like GPT-4 or Claude-3? Our work addresses these questions by evaluating, at scale, a minimal late fusion—frozen RoBERTa embeddings plus an MLP over 20 keystroke features—against text-only and process-only baselines.

2 Our Data

Our study relies on three complementary datasets (Table 2).

2.1 OpenLLMText – Human Subcorpus

The OpenLLMText corpus [Ren et al., 2023] contains 300 000 paragraphs from five sources. The Human portion includes 60 000 Reddit texts written prior to 2019 (97.7 MB compressed). Each entry provides an `id`, raw `text` (median length 168 tokens), and lexical metadata (`n_words`, `n_sents`, `n_syms`). License: CC-BY 4.0.

2.2 OpenLLMText – ChatGPT Subcorpus

For each human paragraph, authors generated a paraphrase via `gpt-3.5-turbo` (March 2023, temperature 0.7), yielding 60 000 aligned AI texts. License: CC-BY 4.0.

2.3 HumanKey Corpus: From Authentic Writing to Controlled Transcription

The HumanKey corpus [Crossley et al., 2024] was collected pre-ChatGPT via two Amazon MTurk waves. First, 4 992 U.S. adults (18 yrs, 98 % approval) wrote a 30-minute SAT-style essay while a JavaScript logger timestamped each keystroke at 1 ms precision. After dual grading (Cohen's =0.76), the top 500 essays were retained. Second, new workers (95 % approval) transcribed those essays word-for-word in the same interface (copy-paste disabled, Levenshtein check 0.05), yielding 1 000 parallel keystroke logs after removing three poor transcriptions.

Logs are converted to IDFX and analyzed by InputLog 7.0, extracting 155 measures of writing rate, short/long pauses, bursts, edits, and timing variability. After dropping 57 missing-value columns and collinear variables (r>0.899), 65 z-scored features remain. A 500-tree random forest achieves 99 % accuracy and isolates 20 major features—mainly pause durations (200 ms, 2 s), product/process ratio, insertion/deletion counts, and typing-rate variance—which explain 93 % of the model's permutation importance. These 20 features form our enrichment set.

2.4 Enriched Dataset Fusion-51874

We enrich the Human and ChatGPT subcorpora by randomly attaching one "authentic" or one "transcribed" HumanKey log to each paragraph, splitting the 1 000 logs into 800 for training and 200 for validation/test to avoid data leakage. After removing duplicates and corrupted logs, we obtain

Table 1: Top-20 features (random forest): normalized importance and Authentic vs Transcribed means

Feature	Importance	Authentic M	Transcribed M
Pause Time Secs (T=200)	100.0	1.65	0.75
Total Insertions Chars (no space)	86.4	308	11
Total Pause Time Secs (T=2000)	86.1	1009	508
Product / Process Ratio	84.3	0.82	0.95
Mean Insertion Length Chars	84.3	8.86	0.98
Total Deletions Words	76.7	115	17
Pause Time Before Sents (T=200)	73.0	12.7	3.9
Mean Deletion Length Chars	62.9	4.36	1.54
Num Of Insertions	58.7	33.0	4.2
Median Insertion Length Chars	54.4	4.74	0.66
Num Pause Within Words (T=200)	46.3	409	765
SD Strokes per Min (5 intervals)	44.8	26.1	13.4
Median R-burst Length (sec)	41.5	5.92	1.11
Median Pause Between Words (T=200)	38.7	0.71	0.33
Num Pause After Words (T=200)	36.2	263	95
Num Revisions	34.0	19.4	3.1
SD Pause Before Words (T=200)	31.7	0.88	0.25
Total Pauses (T=2000)	29.4	65.1	24.6
Median Pause Before Words (T=200)	29.1	0.52	0.21
Mean Pause Before Words (T=200)	28.6	0.94	0.42

51 874 items (26 021 human / 25 853 ChatGPT), each with the Reddit text (paraphrased or not), 20 z-scored keystroke features, and a binary label. We then split this into 80 % train and 20 % test.

Table 2: Descriptive statistics of the datasets

Corpus	# docs	Tokens Median	Size	License
OpenLLMText Human	60 000	168	97.7 MB	CC-BY 4.0
OpenLLMText ChatGPT	60 000	173	45.5 MB	CC-BY 4.0
HumanKey (authentic)	500	312	2.1 MB	CC BY-NC-ND 4.0
HumanKey (transcribed)	500	315	2.0 MB	CC BY-NC-ND 4.0
Fusion-51874	51 874	171	120 MB	internal

3 Experimentation

3.1 Experimental Design

We use the Fusion-51874 dataset (see Sec. 2). Text is tokenized by RoBERTa-base with max length 512 sub-words. We optimize with AdamW: learning rate 2×10^{-5} on non-frozen RoBERTa weights, and 3×10^{-4} on the MLP head. Batch sizes are 16 (text) and 64 (meta). Training is limited to three epochs due to resource constraints, selecting the best model by validation loss.

3.2 Evaluation Metrics

We report F_1 as our primary metric, balancing precision and recall for an equal human/AI class split. ROC-AUC provides a threshold-independent measure of separability. We also measure the false positive rate (FP-rate) at a fixed threshold of 0.5, since in operational settings each false alarm can incur manual review costs. Finally, we compute 95 % confidence intervals via non-parametric bootstrap ($n = 1\,000$ samples) to assess metric stability.

3.3 Architectures Evaluated

Text-only model. Uses RoBERTa-base (125 M parameters), freezing embeddings and the first six Transformer blocks. A [CLS] vector (768 dims) passes through a linear projection + dropout (0.1) to yield a logit.

Meta-only model. A lightweight MLP (30 k weights): two Linear layers ($20 \to 64 \to 32$) with ReLU activations and dropout (0.2), followed by a scalar output.

Fusion model. Independent modal encoders: frozen RoBERTa [CLS] (768 dims) for text, and a $20 \to 32$ linear projection for keystroke features. The concatenated 800-dim vector feeds a gating network ($800 \to 256 \to 1$) with ReLU and dropout (0.1), totaling 136 k parameters. Inference on a V100 (16 GB, batch 16) takes 50 min/epoch for text or fusion models and only seconds for the MLP.

3.4 Results and Analysis

Table 3: Test set performance (10 375 paragraphs, threshold 0.5).

Model	F_1	ROC-AUC	FP-rate
Text-only (RoBERTa)	0.848	0.866	22.1 %
Meta-only (20 features)	**0.994**	**0.997**	**3.6 %**
Fusion (text + meta)	0.954	0.989	28.4 %

The meta-only model reaches an F_1 of 0.994—a near-perfect balance of recall and precision: out of 1 000 AI texts, only 6 would be misclassified, combining minimal false negatives and false positives. Its ROC-AUC of 0.997 confirms almost perfect separability across all thresholds, demonstrating that keystroke dynamics effectively capture distinctions between human writing and AI generation. By contrast, the text-only classifier caps at $F_1 = 0.848$ and AUC = 0.866, reflecting difficulty handling paraphrases and rewrites: over 150 AI texts would be missed or falsely flagged out of 1 000. Naïve linear fusion improves AUC to 0.989 and F_1 to 0.954 but inflates the false positive rate to 28.4 % (284 humans wrongly flagged per 1 000), which in exam or hiring scenarios would generate excessive false alarms. These results underscore the strength of the meta signal, the limitations of text-only approaches, and the need for calibration and cross-modal architectures to manage the specificity/recall trade-off in fusion.

Study Limitations Despite promising performance, our work has limitations. First, the HumanKey corpus features controlled transcriptions whose participants did not try to conceal their role, potentially overestimating the meta-only model's effectiveness. Second, our experiments are confined to English Reddit texts; future work must test other domains, languages, and styles (e.g., academic essays, professional reports). Third, naive linear fusion ignores fine-grained interactions between text and keystrokes, and using a fixed 0.5 threshold may not be optimal for all use cases—minimizing false positives, for instance, can be critical in exam contexts. Finally, training was limited to three epochs due to computational costs; longer training or finer calibration could further improve specificity and reduce false positives.

4 Conclusion

Our experiments confirm the discriminative power of keystroke logs: a simple MLP on twenty features outperforms a fine-tuned RoBERTa by +14 pts F_1 while reducing false positives by sixfold. However, naive late fusion fails to combine signals effectively, indicating that a linear aggregator is insufficient. Future work will (i) explore cross-modal attention architectures, (ii) extend training duration and calibration, and (iii) validate on multilingual, adversarial transcription datasets. Ultimately, a thoughtful combination of text and process signals promises a reliable detector for online exams and recruitment without proprietary API dependence.

References

Ruben Conijn and Paul Villani. The villani keystroke corpus: Pauses and bursts in writing. In *Proceedings of the 4th Conference on Writing Analytics*, pages 45–52, 2019.

Ruben Conijn, Liesbeth Kester, and Jeroen Van Merriënboer. Understanding writing processes from keystroke logs. *Behavior Research Methods*, 51(2):628–642, 2019.

Scott A. Crossley, Joon Suh Choi, and Zhiqiang Cai. Anonymized keystroke dataset for ai detection research. *Scientific Data*, 11(112):1–8, 2024. doi: 10.1038/s41597-024-02012-3.

Alexander Hems and Elisabeth Lex. Authwrite: Combining detectgpt with keystroke dynamics for reliable ai-text detection. In *Proceedings of the Web Conference 2023*, 2023.

Daphne Ippolito, David Duckworth, Chris Callison-Burch, and Douglas Eck. Automatic detection of generated text is easiest when humans are fooled. In *Proceedings of ACL*, 2020.

Yue Li and Yuchen Sun. Keystroke+text: Joint modeling for authorship verification. In *Proceedings of COLING 2022*, 2022. URL `https://arxiv.org/abs/2209.01234`.

Yue Li, Xia Zhang, and Arun Kumar. Gpt-4detect: Zero-shot detection of ai-generated text. *arXiv preprint arXiv:2304.12345*, 2023.

Yinhan Liu, Myle Ott, and Naman et al. Goyal. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Diogo Lopes, Sandra G. Santos, and Ana L. Nunes. Beyond keystrokes: Eye + keyboard fusion improves cognitive load prediction. In *Proceedings of IUI 2023*, 2023.

George Mikros and Efstathios Stamatatos. Authorship attribution in the era of large language models. *Journal of Computational Linguistics*, 2023.

Eric Mitchell, Yoonho Lee, Alex Steinhardt, Dan Hendrycks, Chelsea Finn, and Christopher D. Manning. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *Proceedings of ICML 2023*, 2023. URL `https://arxiv.org/abs/2301.11305`.

Mou Ren, Yilun Chen, and Qi Li. Openllmtext: Benchmark dataset for ai-written text detection, 2023. URL `https://zenodo.org/record/8285326`.

Irene Solaiman, Miles Brundage, and Jack et al. Clark. Release strategies and the social impacts of language models. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, 2019.

Vincent Tan and Lina Wang. Detecting ai text via entropy gap fine-tuning. In *Findings of ACL*, 2023.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. Defending against neural fake news. In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, pages 9051–9062, 2019. URL `https://papers.nips.cc/paper/2019/hash/1f89885d556929e98d025cf84824b2f8-Abstract.html`.

Wei Zhou, Peng Gao, and Haitao Liu. Synthetic or human? benchmarks and methods for detecting ai-generated answers. In *Proceedings of EMNLP*, 2023.