Can a Language Model Write Like an Economist?

Fatih Kansoy*
University of Oxford

This version: November 2026
Latest version: fatih.ai/ai_economist.pdf

Abstract

Language models now help economists revise papers, yet outside judges still tell model prose apart from authors' text even after detailed style rules. This paper asks whether an open model can learn disciplinary prose directly from economists' paragraphs without explicit rules. We train an open model in two stages on one GPU, first on paired drafts and authors' paragraphs from pre-2023 papers and then against a co-trained judge whose authorship probability supplies a length-adjusted reward. On introductions from working papers dated 2026, two outside frontier judges take its revisions for the authors' text about half the time or more, far above raw drafts and a frontier model with our writing guide. Across two runs, the adversarial step adds a large gain that holds at nearly equal length. A small human reading test points in the same direction. Independent content checks show a clear trade-off because trained writers alter claims more often than the guided model. Examples and adversarial feedback bring an open model much closer to economists' prose than our writing guide, though research use still requires an author check on meaning.

Keywords: Large language models; Academic writing; Adversarial training; Reinforcement learning; Economics prose

JEL classification: A14, C45, C88

Data and code. The authors' paragraphs come from other researchers' papers and cannot be shared in full. The prompts are printed in the Online Appendix. The code for training, judging, content checks, and replication, together with the anonymised evaluation data that produce every table and figure in the paper, is publicly available at github.com/kansoy/ai-economist.

*University of Oxford. fatih.kansoy@economics.ox.ac.uk.

1. Introduction

Ask a frontier language model which of two economics paragraphs reads better and it usually picks a model text over what economists wrote. Ask the same model which paragraph the economists wrote and it spots the authors almost every time. Even when Claude Opus 5.5 uses a detailed writing guide, a cliché list, and three published examples to revise an AI draft with identical facts, an outside judge still identifies the authors in 119 of 120 comparisons. Explicit style rules fail to erase the machine signature. We show that an open model trained on economists' paragraphs and against a co-trained judge comes much closer to their prose than the guided frontier model we tested and is taken for the authors about half the time or more on 2026 working papers. Because the trained model alters factual claims more often than the prompted model, its revisions still need an author check.

This gap matters as language models enter daily drafting. Korinek (2023) rates text editing among the most practical uses of generative AI for economists but notes that readers often find model prose sterile. Economics carries rhetorical conventions (McCloskey 1983) that generic models smooth away. Asked why it separated guided revisions from published paragraphs, our judge named concrete disciplinary habits. It replaced technical terms with informal substitutes, added bridges economists leave implicit, and flattened uneven phrasing. For example, Megginson and Netter (2001) write that many privatised firms in Chile were renationalised during the early-1980s debt crisis, yet Opus wrote "came back into state hands" despite rules to keep field terms. A useful assistant must preserve these habits without error in factual claims.

We ask whether or not an open model can learn economists' prose from their paragraphs alone without style rules. Most economists absorb disciplinary style from papers rather than checklists. In our fixed-content task, an open model turns an AI draft into prose that outside judges take for the authors while we also measure how often revision alters claims because authentic prose that shifts a result is unsafe. We adapt Bou et al. (2026) so the model learns first from draft-author pairs and then against a co-trained judge that separates revisions from originals.

To answer both questions, we build a large dataset for fixed-content revision. Our corpus covers 2,120 pre-LLM economics and finance papers (1995 to 2022), mostly open working papers before copy-editing, from which we build 5,267 screened pairs matching an AI draft with the authors' paragraph. Fixed-content revision is essential. In early trials from notes, model and human texts diverged enough in substance that judges could decide on facts alone. To guard against memorisation, our test benchmark combines 137 introduction paragraphs from 2026 NBER working papers (written after the base model's January 2025 cut-off) with 54 recall-screened paragraphs from 1999 to 2022.

Our method runs in two stages on one GPU with 4-bit Gemma 4 31B and low-rank adapters. Step 1 fine-tunes the writer on the 5,267 pairs to learn pre-2023 prose; Step 2 updates it by adversarial policy optimisation against a co-trained judge whose length-adjusted authorship probability supplies the stylistic reward. The writer is never scored by its own judge. Two outside frontier models from distinct families, GPT-6.1 Sol and Claude Opus 5.5, read every test pair in both orders and judge authorship alone. Two blind annotators audit every revision for content changes. Human readers also take a small authorship test.

Both training stages move outside judgements far more than the writing guide we tested. On 2026 papers, neither judge often mistakes the raw draft for the authors; guided Opus gains substantially only under GPT. Supervised training lifts both judges sharply before adversarial updates move GPT-6.1 Sol to a clear majority and Opus to roughly half (about 21 percentage points across all 191 paragraphs and both judges). A second seed repeats this gain closely on revisions of nearly equal length. Two limits bound this finding. On older pre-2023 papers every system scores far lower, although trained writers remain well ahead of both prompted baselines. Trained writers also alter claims more often than guided Opus under both annotators; even when every major change counts as a failure, run 1's gain stays clearly positive. Human readers likewise spot the authors less often against trained writers than against the draft, though that small sample cannot separate the two writers.

These findings indicate that many conventions of economic prose are tacit and easier for a trained model to absorb from examples and adversarial feedback than for a prompted model to follow from rules. Our test compares two complete systems rather than examples against rules alone. It links to three strands of research. First, the supervised stage builds on paraphrase inversion (Krishna, Wieting, and Iyyer 2020, Rivera Soto, Chen, and Andrews 2025), where an inversion model learns to undo a paraphraser's habits from modest data. Second, the adversarial stage adapts Bou et al. (2026), who train a 32B writer against a 235B judge on NIH grant abstracts on a cluster. We run both on one 4-bit 31B model on one GPU and add post-cut-off tests, length control, two-seed replication, pilot diagnostics, and a content audit. Unlike Li et al. (2026), who train scientific writers against quality rubrics, we score authorship alone and audit claims separately. Third, because judges show position, length, and self-recognition bias (Zheng et al. 2023, Dubois, Liang, and Hashimoto 2024, Panickssery, Bowman, and Feng 2024), we score both orders with two outside families distinct from Gemma and remove length effects from the reward.

Two scope limits remain. Our main tests measure whether two outside models attribute one introduction paragraph to authors rather than whether the prose is better. Dates and recall probes limit contamination risk without proof of zero exposure. Section 2 presents the corpus, paired pools, and test sets. Section 3 sets out training and evaluation. Section 4 reports the results and limits before Section 5 concludes.

2. Data

To test whether or not a model can learn economists' prose from examples, we require a large sample of genuine economic writing produced before modern language models entered economics. Our corpus holds 2,120 economics and finance papers (1995 to 2022) across five journal groups (top-five economics, other leading economics, finance, economic history, and survey journals). That pre-2023 cut-off secures a human baseline untouched by modern models. The five groups cover varied empirical and theoretical conventions. Most come from open working-paper archives rather than publisher typesets; 754 come from `nber.org` and at least 1,128 from working-paper servers (NBER, central banks, IZA, EconStor, MPRA, and arXiv), with the rest from publishers and universities. This provenance matters. Working papers record how economists themselves write and circulate research before copy-editors impose a house style.

Our question also requires a task that separates writing style from content choice. In early trials where frontier models wrote from notes, machine and human paragraphs diverged enough in factual detail that judges could decide on substance alone. Fixed-content revision removes that confound because every AI draft must preserve every argument, number, name, and citation of the authors' paragraph. From 1,900 corpus papers (aside from 60 used in guide trials), filters extract 7,866 paragraphs across all sections (introductions, data, methods, results, and conclusions). Claude Haiku 5.5 retains 6,118 complete prose paragraphs and drops tables, equations, and debris. For each kept paragraph, GPT-6.1 Sol or Claude Sonnet 5.5 (split evenly to prevent overfitting one drafter) writes an AI draft that keeps every fact. Automated checks drop 492 drafts that altered a number or citation and 201 near-copies (similarity above 0.8). That screen leaves 5,425 verified pairs from 1,742 papers; minor PDF flaws in 621 target paragraphs were repaired without wording changes (similarity at least 0.95). A paper-level split assigns 5,267 pairs from 1,692 papers to training and 158 pairs from 50 papers to validation (Table 1, Panel A). Each prompt gives the draft together with the journal, section, and last 60 words of the prior paragraph for context.

Adversarial training needs fresh paragraphs that the supervised writer never saw. If the internal judge trained on Step 1 paragraphs, partial memorisation could make revisions appear artificially human and corrupt the reward. From corpus paragraphs outside the supervised pairs and test papers, we build 1,429 new pairs drafted by GPT-6.1 Sol and revised by the supervised writer at temperature 0.7. A paper-level split allocates 1,225 pairs to train the judge before Step 2. The other 204 held-out pairs test the judge and then enter the adversarial pool (Table 1, Panel A), joined by 600 drafts from 51 open-access articles of *Brookings Papers on Economic Activity* (1999 to 2022) for 804 unseen drafts.

Fair evaluation must balance two opposite threats. Older papers may sit in pre-training weights; recent working papers may carry AI editing. Because large models can recall text seen once in

TABLE 1. Data for training and testing

Sample	Paragraphs	Papers	Use
Panel A. Training			
Corpus, 1995–2022		2,120	Source of the authors' paragraphs
Supervised training pairs	5,267	1,692	Writer, step 1
Validation pairs	158	50	Writer, step 1
Judge training pairs	1,225		Judge before step 2
Adversarial training drafts	804		204 from the corpus, 600 from Brookings Papers
Panel B. Testing			
NBER working papers, 2026	137	59	Written after the base model's cut-off
Papers of 1999–2022	54	53	Never trained on, not recalled word for word

Notes: Every pair holds an AI-style draft of a paragraph and the authors' own paragraph. Every draft keeps all numbers of the authors' paragraph. The corpus papers were mostly read in their working-paper versions. Training pairs come from all sections of the papers; test paragraphs are all from introductions. The Brookings Papers drafts come from 51 articles of Brookings Papers on Economic Activity that neither the writer nor the judge had seen before the adversarial step. The older test set lists one paragraph twice, so it holds 53 distinct paragraphs.

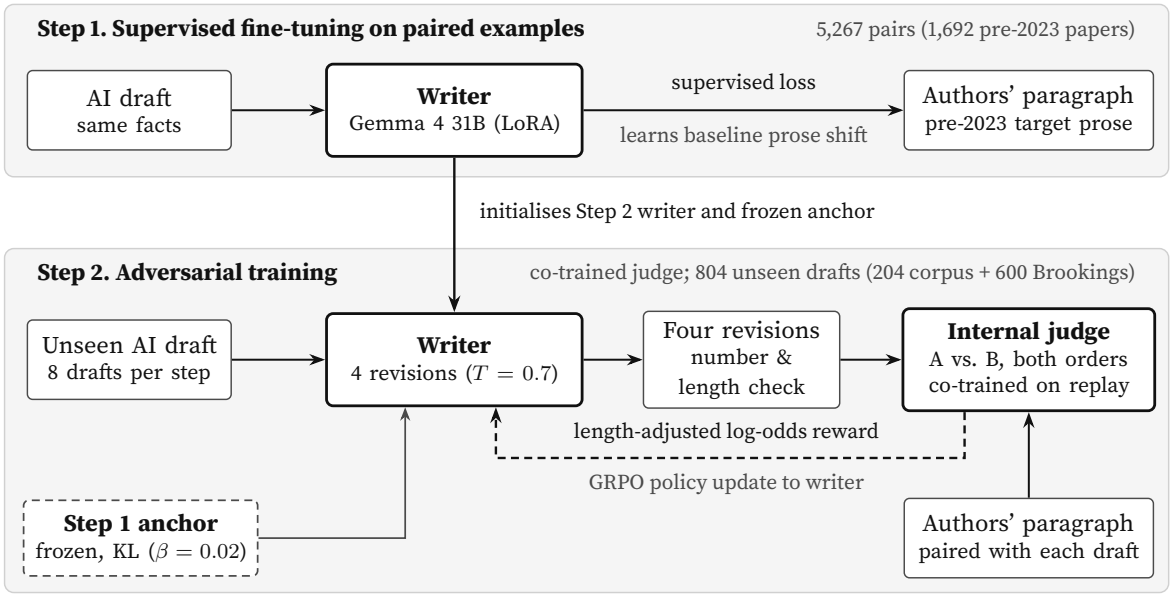
pre-training (Carlini et al. 2021) and Gemma 4 31B's pre-training ends in January 2025 (Google DeepMind 2026), text written after that date cannot sit in its base weights (Shi et al. 2024). Our primary test set draws 137 introduction paragraphs from 59 NBER working papers dated January to November 2026 across finance, economic history, and other core programmes with no author repeated (Table 1, Panel B). Every paragraph passed length, layout, and prose filters, matched a second PDF extraction, and was read by eye to fix broken hyphens without wording changes. Earlier circulation before January 2025 was not checked.

To complement the 2026 papers with guaranteed pre-LLM prose, our second test set holds 54 introduction records from 1999 to 2022. One paragraph appears twice under two OpenAlex IDs to preserve our 98-paragraph pilot benchmark; the set thus holds 53 distinct paragraphs from 53 papers, clustered as one paper in bootstrap intervals. Each older candidate entered only after a recall probe confirmed that no model could continue it word for word from its title and opening half, though subtler familiarity remains possible and other paragraphs of some older papers sit in training. All 191 test drafts come from GPT-6.1 Sol so every system starts from the same input. No test paragraph enters training or judge pairs; no adversarial draft shares a test paper; both runs score all 191 records (among them the 98 pilot paragraphs).

3. Method

Our procedure runs in two stages on one rented A100 GPU (Figure 1). We load 4-bit instruction-tuned Gemma 4 31B (Dettmers et al. 2023) and attach three low-rank adapters (Hu et al. 2022) for the writer, a fixed Step 1 anchor, and the internal judge. Step 1 fine-tunes the writer over two passes on the 5,267 pairs (660 updates, 4 hours and 39 minutes) for the supervised baseline.

FIGURE 1. The two training steps



Notes: Step 1 fine-tunes the writer on 5,267 pairs of an AI draft and the authors' paragraph. Its weights initialise both the Step 2 writer and the frozen KL anchor ($\beta = 0.02$). Before Step 2, the judge trains alone on 1,225 clean pairs. In Step 2, the writer draws four revisions per unseen draft at temperature 0.7 and the judge compares each with the authors' text in both orders. Length-adjusted log-odds give the GRPO reward while the judge updates on replay unless accuracy exceeds 0.90; writer updates pause below 0.60. All three LoRA adapters share one 4-bit Gemma 4 31B base on one GPU.

In Step 2, the writer trains adversarially against the internal judge, grounded in adversarial networks (Goodfellow et al. 2014), imitation learning (Ho and Ermon 2016), and policy-gradient text generation (Yu et al. 2017). Because adversarial text training works best from pre-trained models (Wu, Li, and Yu 2021), the judge first trains alone on the 1,225 clean pairs. At each step, the judge compares revision y with authors' text $y^\star$ for prompt x (A and B). To cancel position bias (Zheng et al. 2023), we average authorship probability across both orders in

$$D_\phi(x, y, y^\star) = \tfrac{1}{2}\big[P_\phi(\mathrm{A} \mid x, y, y^\star) + P_\phi(\mathrm{B} \mid x, y^\star, y)\big]. \tag{1}$$

Each step draws $K = 4$ revisions for $P = 8$ drafts (four per pool) at temperature 0.7. Valid revisions earn the clipped log-odds (± 8) of Equation (1); any revision that alters a number, falls outside 0.7 to 1.4 times draft length, or fails to finish gets -10. Because model judges favour longer passages (Dubois, Liang, and Hashimoto 2024) and authors wrote the longer text in 71 per cent of held-out pairs, we remove length effects from reward r_{jk} within each group via

$$\tilde{r}_{jk} = r_{jk} - b_t\big(\log \ell_{jk} - \overline{\log \ell_j}\big), \tag{2}$$

where ℓ_{jk} is word count and b_t is the within-group slope on log length pooled with decay 0.9.

Writer parameters update through group-relative policy optimisation (Shao et al. 2024) without a value model so all adapters fit on one GPU. With A_{jk} equal to $\tilde{r}_{jk}$ from Equation (2) standardised within its group, each step minimises

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{(j,k):A_{jk}\neq 0} \sum_{t} [A_{jk} \log \pi_\theta(y_{jk,t} \mid x_j, y_{jk,<t}) - \beta\, \kappa_{jk,t}], \tag{3}$$

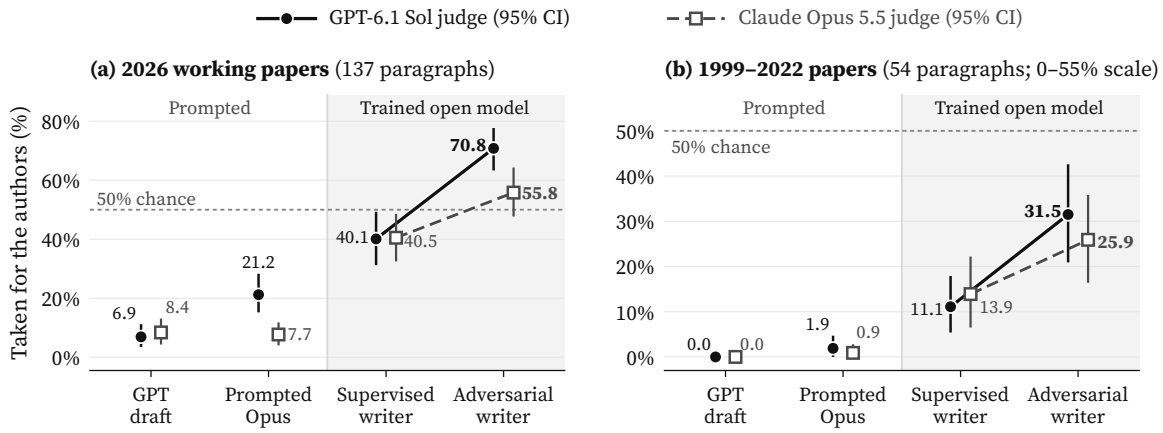
where N counts tokens across non-zero-advantage revisions and $\kappa_{jk,t}$ is per-token KL divergence from the anchor ($\beta = 0.02$). The judge updates on replayed pairs except above 0.90 accuracy (as in one task of Bou et al. (2026)). Below 0.60, writer updates pause until accuracy hits 0.65 so the writer does not chase noise; 15 paused steps end the run (Table A1 in the Online Appendix).

Evaluation stays independent of the internal judge. At steps 0, 50, and the final step, the writer revises all 191 test drafts. Two outside models from distinct families, GPT-6.1 Sol and Claude Opus 5.5, read the title and both paragraphs in both orders and pick the authors' text. Our outcome is the share taken for the authors (chance is 50 per cent), with 95 per cent paper-clustered bootstrap intervals. Five pre-registered rules define success. The final writer must raise overall attribution across all 191 paragraphs and both judges, gain at least 5 points per judge on 2026 papers, keep numbers in at least 90 per cent of 2026 paragraphs, show no fall on older papers, and avoid length-driven gains. Two blind annotators later audited content changes across all revisions (Section C in the Online Appendix).

4. Results

Our trained writers move outside judges far more than a frontier model prompted with rules (Figure 2). On 2026 papers, GPT-6.1 Sol and Claude Opus 5.5 take the GPT draft for the authors in 6.9 and 8.4 per cent of comparisons. With our writing guide, three examples, and one rule pass, Opus reaches 21.2 per cent under GPT yet stays at 7.7 per cent under Opus itself, a pattern that reflects self-recognition (Panickssery, Bowman, and Feng 2024). Earlier guide trials confirmed the same pattern. When Opus wrote 40 paragraphs from notes, GPT-6.1 Sol rated guided text better than the authors in 80 per cent of comparisons (as in Li et al. (2026)) yet never attributed it to them; quality judgements can reward the smoothness that exposes machine prose. On 60 older fixed-content revisions, the guide moved three-syllable words towards the authors (26.2 to 24.6 per cent against 23.7 per cent), yet GPT-6.1 Sol still picked the authors in 119 of 120 comparisons. Supervised training lifts both judges to about 40 per cent on 2026 papers (40.1 and 40.5 per cent) before adversarial updates raise them to 70.8 and 55.8 per cent in run 1. On 1999 to 2022 papers every version scores far lower, although trained writers remain well above the draft (0 per cent) and guided Opus (1.9 and 0.9 per cent).

FIGURE 2. How often each version is taken for the authors' paragraph



Notes: Share of showings in which an outside judge picks the machine version as the authors' paragraph (both orders, so random guessing yields 50 per cent), with 95 per cent paper-clustered bootstrap intervals. Panel (b) uses a zoomed 0 to 55 per cent vertical axis so pre-2023 differences stay clear alongside the 50 per cent line. Shaded areas mark the two trained open-model stages of run 1; the draft and prompted Opus (Claude Opus 5.5 with guide and three examples) are separate baselines across the same 191 paragraphs.

Adversarial training adds about 21 percentage points in both runs (Table 2). Across all 191 paragraphs and both judges, attribution rises from 32.5 to 53.5 per cent in run 1 (+21.1 points, 95 per cent interval 15.4 to 26.6). Run 2 repeats training under a fresh seed and adds 20.8 points (30.4 to 51.2 per cent, interval 15.6 to 26.0). Both runs pass all five pre-registered rules; most gains arrive by step 50. On 2026 papers, however, the two judges diverge sharply. GPT-6.1 Sol picks the final writer over the authors in 70.8 and 68.2 per cent of comparisons, a reversal that shows GPT still sees a systematic difference. Claude Opus 5.5 moves roughly half as far to 55.8 per cent (47.7 to 64.3) and 54.4 per cent (46.3 to 62.5); both intervals include 50 per cent. Length does not drive these gains. Mean length relative to the draft rose only 1.4 and 0.9 per cent; on paragraphs within 5 per cent of draft length, gains stay at 21.4 points (124 paragraphs) and 19.3 points (135 paragraphs). Both final writers kept every number in all 137 paragraphs from 2026, against 85.2 per cent (46 of 54) in run 1 and 90.7 per cent (49 of 54) in run 2 on older papers.

Because numbers are only part of meaning, two blind annotators checked every version against its draft for dropped or added claims, changed hedges, entity errors, and meaning shifts (Table A2 in the Online Appendix). Strictness differs sharply. GPT-6.1 Sol flags 20.9 per cent of authors' paragraphs as major changes; Claude Opus 5.5 flags 0.5 per cent. Both show that trained writers alter content more often than guided Opus (28.3 and 6.8 per cent supervised against 8.4 and 0.5 per cent guided). Against run 1's supervised revisions, adversarial updates add 4.2 and 3.1 points in run 1 (GPT's upper bound hits 12.3 and fails our no-cost rule) and 0.0 and -2.1 in run 2 (whose supervised outputs were not annotated). When every major flag counts as a failure, run 1's gain falls from 21.1 to 14.1 points under GPT and 19.1 under Opus, both clearly positive. Claude Opus 5.5 (one annotator) re-read 30 random flagged revisions with both flags visible. All 30 marked real

TABLE 2. Outside judges before and after adversarial training

Test set	Judge	Taken for the authors (%)			Final minus step 0	
		Step 0	Step 50	Final	Points	95% interval
Panel A. Run 1 (seed 20261115; final writer after 57 updates)						
2026 papers (137)	GPT-6.1 Sol	40.1	67.2	70.8	+30.7	(23.0, 38.3)
	Claude Opus 5.5	40.5	56.9	55.8	+15.3	(7.7, 23.4)
1999–2022 papers (54)	GPT-6.1 Sol	11.1	28.7	31.5	+20.4	(9.8, 32.1)
	Claude Opus 5.5	13.9	18.5	25.9	+12.0	(3.6, 21.3)
All 191	Both	32.5		53.5	+21.1	(15.4, 26.6)
Panel B. Run 2 (seed 20261116; final writer after 54 updates)						
2026 papers (137)	GPT-6.1 Sol	39.1	60.2	68.2	+29.2	(22.6, 36.1)
	Claude Opus 5.5	41.2	50.0	54.4	+13.1	(5.7, 20.4)
1999–2022 papers (54)	GPT-6.1 Sol	4.6	13.9	25.9	+21.3	(11.8, 31.8)
	Claude Opus 5.5	6.5	15.7	25.0	+18.5	(7.4, 30.0)
All 191	Both	30.4		51.2	+20.8	(15.6, 26.0)
Panel C. Paragraphs whose length changed by at most 5%, both judges						
Run 1 (124 paragraphs)					+21.4	(14.8, 28.1)
Run 2 (135 paragraphs)					+19.3	(13.4, 25.2)

Notes: Share of cases in which the outside judge takes the writer's revision for the authors' paragraph. With each pair shown in both orders, a judge that guesses at random reaches 50 per cent. Step 0 is the supervised writer. "Final" is the test at the step where the run stopped (72 in run 1, 69 in run 2), with the last writer updated before the judge's accuracy fell below 0.60. Changes are paired differences with 95 per cent bootstrap intervals over papers.

differences; Opus judged that 14 altered claims ("are followed by" became "precipitates" in one paragraph and a relation reversed sign in another).

Adversarial training worked only after two failed pilots showed two rules (Table A3 in the Online Appendix). First, the judge must already separate human from model prose before Step 2. In pilot 1, a fresh judge trained online on about 64 pairs per step while the writer sampled at 1.0. Accuracy peaked at 0.65 and fell to 0.51; that noisy reward drove the writer backwards under both judges (-12.7 and -10.8 points on 51 paragraphs from 2026). Once trained alone on 1,225 clean pairs, the same judge reached 0.990 on held-out pairs (0.968 on 47 older paragraphs and 0.892 on 51 from 2026). Simple cues explained part of that gap. Authors wrote the longer text in 71.3 per cent of held-out pairs and a naive Bayes model reached 0.824; authors favoured `note`, `fact`, `indeed`, and `whereas` while the supervised writer preferred `examine`, `identify`, and `assess`.

Second, training must sample at the temperature that trained the judge. Even with that judge, pilot 2 collapsed in five steps (0.63 to 0.34). A diagnostic showed that the judge had learned on revisions at 0.7 while pilot 2 sampled at 1.0; on identical drafts without updates, accuracy was 0.92 to 1.00 at 0.7 but dropped to 0.29 to 0.46 at 1.0. Once sampling matched 0.7, pilot 2 gained 12.0 points across 98 paragraphs and both main runs worked. Because pilot 1 also used 1.0, these runs show a working setup without a separate test of each rule. Across both runs, judge accuracy fell from

0.88 and 0.94 to or below 0.60 (0.60 after 57 updates in run 1 and 0.59 after 54 in run 2) before each stopped after 15 paused steps (Figure A1 in the Online Appendix).

Human judgements point the same way as model judges, although human detection is noisy (Ippolito et al. 2020, Clark et al. 2021) and scientists miss about a third of generated abstracts (Gao et al. 2023). On an open web page, visitors saw up to five pairs from 24 of the 2026 paragraphs and picked the human text. By 17 November 2026, 23 readers gave 95 answers (16 finished all five pairs; 21 held or pursued a master's or doctorate). Readers spotted the authors in 61.8 per cent of pairs against the draft (21 of 34), against 37.9 per cent for the supervised writer (11 of 29) and 43.8 per cent for run 1's adversarial writer (14 of 32; Figure A2 in the Online Appendix). When we pool both trained writers, detection drops by 20.8 points relative to the draft (95 per cent reader-clustered interval 6.9 to 35.1). That exploratory contrast cannot separate the two writers; accuracy did not rise across pairs despite feedback.

Several limits qualify these findings. First, our evidence rests on two judges; because GPT-6.1 Sol wrote every test draft, it may react more strongly than Opus to revisions of its own prose. A third judge family would help. Second, the reading sample is small and identities were unverified. Third, the task revises one introduction paragraph with fixed content rather than a draft from notes. Fourth, some 2026 papers may carry AI editing; older pre-2023 paragraphs stay far below chance. Fifth, because every draft is a GPT paraphrase, part of the gain may simply undo its habits (Rivera Soto, Chen, and Andrews 2025). Sixth, we compare two full systems rather than examples against rules alone; fixed judges, best-of-N sampling, and many-shot prompting remain for future work. Finally, both runs ended once the 31B judge hit its floor; because Bou et al. (2026) find that larger judges learn better rewards, a larger judge is a natural next step.

5. Conclusion

We find that an open 31B model trained directly on economists' paragraphs comes much closer to their prose than a frontier model guided by our writing rules. Style guides alone did not erase the machine signature. Supervised training of Gemma 4 31B on 5,267 pre-2023 pairs followed by adversarial updates against a co-trained judge lifted attribution to about half or more on 2026 working papers. Both seeds confirm that adversarial training adds roughly 21 points at nearly equal length; in run 1, that gain stayed positive even when major content changes counted as failures.

Two practical lessons follow. Adversarial training works only when the judge already separates human from model prose and scores revisions at its training temperature, both verifiable before a full run. Because trained writers alter claims more often than prompted models, any workflow requires a separate author check. All training ran on one A100 GPU. Total API spending across pilots, drafts, judges, and content checks came to 131 dollars.

References

Bou, Albert, Javier Valdes-Aleman, Jon M. Laurent, Samuel G. Rodriques, Andrew D. White, Michaela Hinks, and Siddharth Narayanan. 2026. "Learning to Reason from Reference Samples Alone: Reinforced Adversarial Learning for Modern LLM Post-Training." arXiv preprint arXiv:2610.38851. https://arxiv.org/abs/2610.38851.

Carlini, Nicholas, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. "Extracting Training Data from Large Language Models." In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–2650. https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

Clark, Elizabeth, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. "All That's 'Human' Is Not Gold: Evaluating Human Evaluation of Generated Text." In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 7282–7296. Online: Association for Computational Linguistics. https://aclanthology.org/2021.acl-long.565/.

Dettmers, Tim, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. "QLoRA: Efficient Finetuning of Quantized LLMs." In *Advances in Neural Information Processing Systems*, edited by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, vol. 36: 10088–10115: Curran Associates, Inc.

Dubois, Yann, Percy Liang, and Tatsunori B. Hashimoto. 2024. "Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators." In *Proceedings of the First Conference on Language Modeling (COLM 2024)*, https://openreview.net/forum?id=CybBmzWBX0. arXiv version (arXiv:2404.04475) also lists Balázs Galambosi as an author.

Gao, Catherine A., Frederick M. Howard, Nikolay S. Markov, Emma C. Dyer, Siddhi Ramesh, Yuan Luo, and Alexander T. Pearson. 2023. "Comparing Scientific Abstracts Generated by ChatGPT to Real Abstracts with Detectors and Blinded Human Reviewers." *npj Digital Medicine* 6: 75.

Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. "Generative Adversarial Nets." In *Advances in Neural Information Processing Systems*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, vol. 27: Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf.

Google DeepMind. 2026. "Gemma 4 Model Card." https://ai.google.dev/gemma/docs/core/model_card_4. Released 31 March 2026; accessed 17 November 2026.

Ho, Jonathan, and Stefano Ermon. 2016. "Generative Adversarial Imitation Learning." In *Advances in Neural Information Processing Systems*, edited by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, vol. 29: Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2016/file/cc7e2b878868cbae992d1fb743995d8f-Paper.pdf.

Hu, Edward J., Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. "LoRA: Low-Rank Adaptation of Large Language Models." In *International Conference on Learning Representations*, https://openreview.net/forum?id=nZeVKeeFYf9.

Ippolito, Daphne, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. "Automatic Detection of Generated Text is Easiest when Humans are Fooled." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1808–1822. Online: Association for Computational Linguistics.

https://aclanthology.org/2020.acl-main.164/.

Korinek, Anton. 2023. "Generative AI for Economic Research: Use Cases and Implications for Economists." *Journal of Economic Literature* 61 (4): 1281–1317.

Krishna, Kalpesh, John Wieting, and Mohit Iyyer. 2020. "Reformulating Unsupervised Style Transfer as Paraphrase Generation." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 737–762: Association for Computational Linguistics.

Li, Tianjian, Chuanyang Jin, Seungone Kim, Tong Chen, John Hoffman, Yoram Bachrach, Pedro Rodriguez, Ping Yu, Eryk Helenowski, Swarnadeep Saha, Jack Lanchantin, Weizhe Yuan, Ilia Kulikov, and Jason Weston. 2026. "Aligning Autoresearch: Co-Improvement between Agents & Humans." arXiv preprint arXiv:2610.38882. https://arxiv.org/abs/2610.38882.

McCloskey, Donald N. 1983. "The Rhetoric of Economics." *Journal of Economic Literature* 21 (2): 481–517. https://www.jstor.org/stable/2724987.

Megginson, William L., and Jeffry M. Netter. 2001. "From State to Market: A Survey of Empirical Studies on Privatization." *Journal of Economic Literature* 39 (2): 321–389.

Panickssery, Arjun, Samuel R. Bowman, and Shi Feng. 2024. "LLM Evaluators Recognize and Favor Their Own Generations." In *Advances in Neural Information Processing Systems*, edited by A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, vol. 37: 68772–68802: Curran Associates, Inc.

Rivera Soto, Rafael Alberto, Barry Y. Chen, and Nicholas Andrews. 2025. "Mitigating Paraphrase Attacks on Machine-Text Detection via Paraphrase Inversion." In *Findings of the Association for Computational Linguistics: ACL 2025*, 4421–4433: Association for Computational Linguistics.

Shao, Zhihong, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models." arXiv preprint arXiv:2402.03300.

Shi, Weijia, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2024. "Detecting Pretraining Data from Large Language Models." In *The Twelfth International Conference on Learning Representations*, https://openreview.net/forum?id=zWqr3MQuNs.

Wu, Qingyang, Lei Li, and Zhou Yu. 2021. "TextGAIL: Generative Adversarial Imitation Learning for Text Generation." In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, vol. 35: 14067–14075: AAAI Press.

Yu, Lantao, Weinan Zhang, Jun Wang, and Yong Yu. 2017. "SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient." In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, vol. 31: AAAI Press.

Zheng, Lianmin, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena." In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, edited by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, vol. 36: 46595–46623: Curran Associates, Inc.

Online Appendix

A. Prompts

Drafts. GPT-6.1 Sol or Claude Sonnet 5.5, with the system message "You are an economist and a careful writer." and the instruction:

> Rewrite the paragraph below in your own words, the way you would naturally write it for an economics paper. Keep every piece of content: every fact, number, year, citation, name, example and qualification, and add nothing. Reply with the paragraph only.

Writer. One user turn, no system message (the section is "introduction" for every test paragraph):

> Journal: {journal} ({journal family}). Section: {section}.
> End of the previous paragraph: {last 60 words of the paragraph before}
> Revise this draft paragraph into the authors' own published prose. Keep every fact, number and citation.
> Draft:
> {draft}

Internal judge. One user turn; the answer is read from the logits of the letters A and B:

> Below are two versions of one paragraph from {the introduction section of a paper in journal}. One was written by the paper's authors and published. The other was produced by a revision tool from a draft. Which one did the authors write? Answer with the single letter A or B.
> Paragraph A: {...} Paragraph B: {...}

Outside judges. GPT-6.1 Sol and Claude Opus 5.5 (low effort), system message "Answer with one letter only.":

> Below are two versions of one paragraph from the introduction of an economics paper. One was written by the paper's authors. The other was written by an AI model. Which one did the authors write? Answer with the single letter A or B.
> Paper: {title} Paragraph A: {...} Paragraph B: {...}

B. Settings and cost

TABLE A1. Training settings

Setting	Value
Base model	Gemma 4 31B (instruction-tuned), 4-bit NF4
Adapters	LoRA, rank 16, $\alpha = 32$, all projection layers of the language model
Supervised step	learning rate 10^{-4}, effective batch 16, two passes, 660 updates, 4 h 39 min
Judge before the adversarial step	learning rate 10^{-4}, 16 showings an update, 308 updates, 132 min
Adversarial step, per step	8 drafts (4 per pool), 4 revisions each, temperature 0.7, at most 400 tokens
Writer update	learning rate 5×10^{-6}, KL weight $\beta = 0.02$, advantages clipped at ± 3
Reward	judge log-odds clipped at ± 8; -10 when a number changes, length is outside 0.7–1.4 of the draft, or the text does not end
Length adjustment	within-group slope on log length, pooled with decay 0.9
Judge update	learning rate 2×10^{-5}, 32 replayed pairs, skipped while running accuracy > 0.90 (at most 20 in a row)
Pause and stop	writer paused below 0.60 until 0.65; stop after 15 paused steps
Test revisions	temperature 0.7, top-p 0.95, at steps 0, 50, and the last step
Hardware	one NVIDIA A100 (80 GB) on Google Colab for each main run (40 GB for the second pilot); peak memory 53.9 GB in run 2
Time	run 1: 72 steps in 7.1 h; run 2: 69 steps in 6.5 h

Notes: Settings from the training code of the main runs and their logs. API spending for the whole study, including the pilots, the drafts, all outside judging, and the content check, was 131.10 dollars.

C. Content fidelity

We fixed the content-fidelity protocol and its decision rule before any annotation ran. Each annotator read the GPT draft as the source alongside one revised version as the edited text, blind to the system that produced it, and logged differences by category together with an overall verdict of same, minor, or major. Our pre-specified rule contained two conditions. Under the first condition, the adversarial stage had to add at most 5 percentage points of major revisions over the supervised writer, with the upper bound of the 95 per cent interval below 10 points for both annotators and both runs. Because the GPT-6.1 Sol upper bound reaches 12.3 points in run 1, that condition is not met. Under the second condition, run 1's adversarial gain had to remain strictly above zero for both annotators when every major revision counts as a failure, a threshold that both annotators pass. Across the five tested versions, the two annotators gave the same major or not-major call on 76 to 92 per cent of paragraphs. Both adversarial runs are compared against the supervised revisions of run 1 because run 2's supervised outputs were not separately annotated. After the main annotation finished, Claude Opus 5.5 re-read 30 randomly selected revisions that at least one annotator had flagged from the supervised and adversarial writers with both annotators' flags in view.

TABLE A2. Content fidelity of each version against its draft

	Annotator GPT-6.1 Sol		Annotator Claude Opus 5.5	
Version compared with the draft	Share	95% interval	Share	95% interval
Panel A. Revisions judged major (% of 191 paragraphs)				
Authors' paragraph (reference)	20.9	(15.2, 26.8)	0.5	(0.0, 1.7)
Prompted Opus with writing guide	8.4	(4.7, 12.9)	0.5	(0.0, 1.7)
Supervised writer	28.3	(21.6, 35.4)	6.8	(3.0, 11.4)
Adversarial writer, run 1	32.5	(25.5, 39.5)	9.9	(5.8, 14.6)
Adversarial writer, run 2	28.3	(21.5, 35.5)	4.7	(1.9, 7.9)
Run 1 minus supervised (points)	$+4.2$	$(-3.4, 12.3)$	$+3.1$	$(-2.6, 9.2)$
Run 2 minus supervised (points)	0.0	$(-7.9, 7.9)$	-2.1	$(-7.3, 2.6)$
Panel B. Taken for the authors, a major revision counted as a failure (%, both judges)				
Supervised writer	22.4		30.6	
Adversarial writer, run 1	36.5		49.7	
Gain (points)	$+14.1$	(8.7, 20.1)	$+19.1$	(13.0, 25.3)

Notes: Major means a dropped or added claim, a changed number, name, or citation, a changed hedge that alters what is claimed, or a change of meaning, as judged by each annotator. The authors' paragraph is the text from which the draft was written; it shows how often two texts meant to hold the same content are judged different. Intervals are 95 per cent bootstrap intervals over papers, paired for differences. Panel B pools the showings of both outside judges, as in Table 2.

D. Pilot and main runs

TABLE A3. Every adversarial run, with the state of the judge

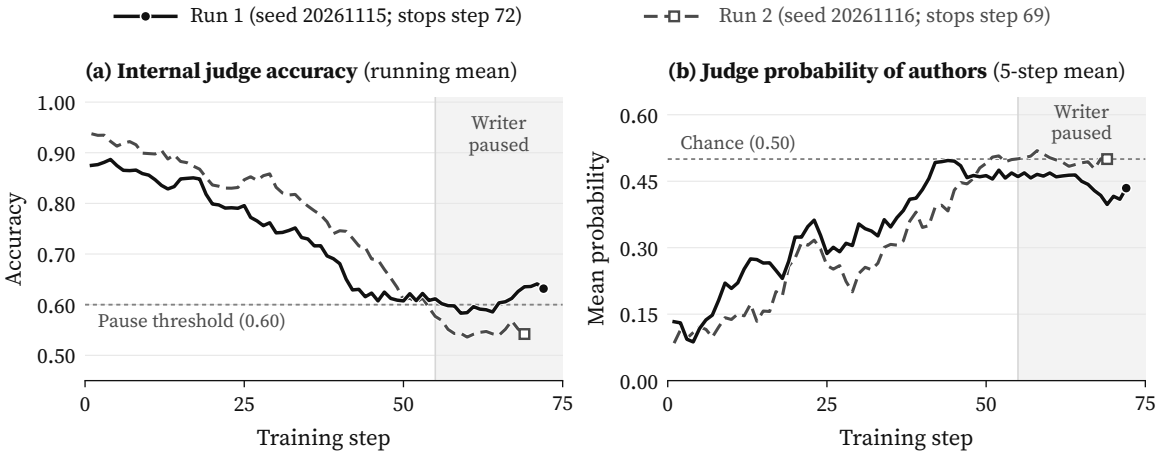
Run	Sampling	Internal judge	Outside judges, change
Panel A. Pilots			
Pilot 1, new judge trained online (61 steps)	$T = 1.0$	0.50 → 0.65 → 0.51	-12.7 (GPT), -10.8 (Opus)[a]
Judge trained alone (308 updates)	$T = 0.7$	0.990 on held-out pairs[b]	no writer training
Pilot 2, trained judge, first try (5 steps)	$T = 1.0$	0.63 → 0.34	stopped
Diagnostic on the same drafts, no training	$T = 0.7$	0.92–1.00	
Diagnostic on the same drafts, no training	$T = 1.0$	0.29–0.46	
Pilot 2, trained judge (34 steps)	$T = 0.7$	0.91 → 0.69	$+12.0$ (5.6, 18.4)[c]
Panel B. Main runs, trained judge			
Run 1 (72 steps)	$T = 0.7$	0.88 → 0.60	$+21.1$ (15.4, 26.6)
Run 2 (69 steps)	$T = 0.7$	0.94 → 0.59	$+20.8$ (15.6, 26.0)

Notes: "Internal judge" is the judge adapter's accuracy at telling the authors' paragraph from the writer's revision, as a running mean at the first step and at the last writer update (per step in the first try of pilot 2), or on held-out pairs where stated. T is the sampling temperature of the training revisions; the judge was trained on revisions sampled at $T = 0.7$. The diagnostic rows score 24 drafts in both orders with no training. Changes are in percentage points with 95 per cent bootstrap intervals over papers.

[a] On 51 paragraphs from 2026 papers; intervals $(-22.5, -2.9)$ and $(-20.6, -1.0)$. [b] 0.968 on the pilots' 47 older test paragraphs and 0.892 on their 51 paragraphs from 2026; the untrained adapter scores 0.41 to 0.48. [c] On 98 test paragraphs with both judges, computed after the verdict; the pilot's rule used its 51 paragraphs from 2026, where the gains were 9.8 and 6.9 points.

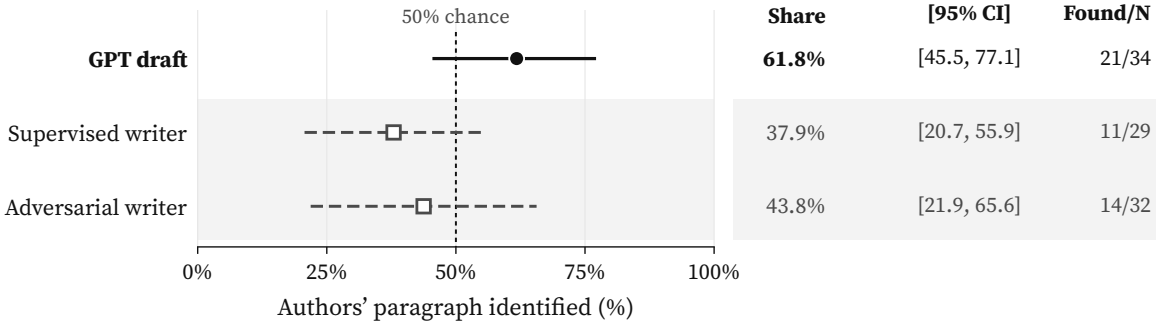
E. Training dynamics and the reading test

FIGURE A1. Training dynamics in the two runs



Notes: Panel (a) shows the internal judge's running accuracy (exponential mean with weight 0.9) on each training step's pairs; writer updates pause whenever accuracy falls below 0.60 (shaded zone from step 55). Panel (b) shows the judge's mean probability that a training revision is the authors' paragraph as a five-step moving mean against the 0.50 chance benchmark. Both panels use zoomed vertical scales; data come from the training logs of both runs.

FIGURE A2. Readers choosing the human paragraph



	Share	[95% CI]	Found/N
GPT draft	**61.8%**	[45.5, 77.1]	21/34
Supervised writer	37.9%	[20.7, 55.9]	11/29
Adversarial writer	43.8%	[21.9, 65.6]	14/32

Notes: Share of pairs in which a human reader picked the authors' paragraph, grouped by the machine version paired against it, with 95 per cent reader-clustered bootstrap intervals and exact shares, intervals, and counts reported on the right. The shaded band marks the two trained writers (the supervised writer at step 0 and the final adversarial writer of run 1) compared with the GPT-6.1 Sol draft across 23 readers, 95 answers, and 24 of the 2026 paragraphs.