# Just Ask: Detecting AI-generated Text by Directly Asking AIs

**Anonymous ACL submission**

## Abstract

Detecting AI generated content is becoming an increasingly important and difficult problem. With each new generation of AI models, models get better at mimicking features of human generated text and evading existing AI detectors. We present JUST ASK, a zero-shot method for detecting AI-generated text that queries a frontier LLM directly with the task of estimating the probability that a given text segment was machine-generated. This method achieves SOTA performance (0.999 AUROC) on representative sampels of AI-detection benchmarks with 0.0% false positive rate exceeding open source detectors' performance and matching proprietary detector performance at more than 70% per token cost reduction. We show that detection ability scales with model capability: across 9 models spanning three families, more capable models are consistently better detectors. Applying JUST ASK at scale, we analyze 48,000 arXiv papers spanning January 2022 to December 2025 and find that AI-assisted text has grown at an exponential rate from at least 0.1% to 9.2%. In arXiv papers, detection rates correlate negatively with author seniority ($r=-0.222$) and vary sharply by field, with Computer Science reaching 15.5% in 2025 while Physics and Mathematics remain below 3%.

## 1 Introduction

The release of ChatGPT in November 2022 marked an inflection point in the accessibility of AI-generated text. Within months, large language models became widely adopted tools for writing assistance across domains ranging from academic research to social media. This rapid adoption raises a fundamental empirical question: how much of the text produced today is written, co-written, or polished by AI and whether we can reliably measure this at scale.

Measuring the prevalence of AI generated text has been a challenge as it requires detectors that are simultaneously accurate, capable of generalizing across different generators and domains, and capable of operating at very low false positive rates, since even a 1% false positive rate can produce faulty predictions in unbalanced distributions.

Existing zero-shot statistical detectors do not meet this bar. We evaluated two leading methods: Binoculars (Hans et al., 2024) and Fast-DetectGPT (Bao et al., 2023): and found that both fail when tested on AI-generated text outside of their benchmark datasets. Some commercial detectors, on the other hand, such as GPTZero (Adam et al., 2026) offer higher performance but are proprietary and can be expensive to operate at scale.

We propose JUST ASK: directly querying a frontier LLM with a prompt that asks it to pretend to be a calibrated detector of AI generated text and then to estimate the probability that a given input text segment was AI-generated. We show that this simple approach is remarkably effective when using current frontier models. We show that detection ability scales reliably with model generation across 9 models spanning three families, more capable models are consistently better detectors of AI generated text. (Table 3, Figure 2).

Applying JUST ASK to academic articles at scale, we produce two main findings:

1. **Temporal acceleration:** In a longitudinal analysis of 48,000 arXiv papers (1,000/month, January 2022 - December 2025), AI-detected prevalence has grown exponentially from at least 0.1% to 9.2% at our conservative threshold (Figure 2).

2. **Demographic patterns:** AI detection rates correlate negatively with author seniority ($r=-0.222$ for max $h$-index; Figure 2) and vary sharply by field: Computer Science reaches 15.5% in 2025, while Physics and Mathematics remain below 3% (Figure 2).

## 2 Related Work

**AI Text Detection.** Zero-shot statistical methods exploit token-level properties of language models for detection. DetectGPT (Mitchell et al., 2023) uses perturbation-based log-probability curvature, Fast-DetectGPT (Bao et al., 2023) replaces this with an efficient analytic approximation. Binoculars (Hans et al., 2024) computes a cross-perplexity ratio between two related models as a proxy of AI-generation probability. On the supervised side, Ghostbuster (Verma et al., 2023) and GPTZero (Adam et al., 2026) train classifiers on labeled data, achieving strong results on benchmarks like RAID (Dugan et al., 2024), MGTBench (He et al., 2023), M4GT-Bench (Wang et al., 2024), and MAGE (Li et al., 2024). Watermarking (Kirchenbauer et al., 2023) offers an orthogonal approach to fingerprint AI-generated text but requires model developer cooperation. Sadasivan et al. (2023) show theoretically that detection becomes harder as models improve. Unlike all of the above, JUST ASK requires no training data, no model internals, and no model developer cooperation.

**Prevalence Estimation.** Prior work has measured AI content spread through indirect lexical signals: excess use of specific vocabulary in peer reviews (Liang et al., 2024b,a), word frequency shifts in biomedical publications (Kobak et al., 2024), and stylistic markers in scholarly literature (Gray, 2024). Others have studied downstream effects, finding that AI-assisted reviews receive higher scores (Russo et al., 2024) with lower confidence and that LLM adoption can vary across societal contexts (Liang et al., 2025; Rao et al., 2025).

## 3 Methods

### 3.1 Just Ask

Our method is straightforward: we prompt a frontier LLM, $M$, with a specific prompt, $p$, and a segment of text, $x$, asking it to return a single number representing a probability between 0.00 and 1.00 that this text segment is AI generated. We interpret its output as a calibrated probability estimate. The full prompt and algorithm are shown in Figure 3 in the appendix.

We then take this probability and apply a threshold $\tau$ to the returned probability: texts scoring at or above $\tau$ are classified as AI-generated. The threshold is selected to minimize False Positives while maintaining Recall as minimizing FPR is critical for prevalence measurement in unbalanced datasets.

### 3.2 Benchmark Evaluation

We evaluated JUST ASK on balanced, random subsets of four established benchmarks: RAID (Dugan et al., 2024) (1,570 texts, 8 domains), HC3 (Su et al., 2023), M4GT-Bench (Wang et al., 2024), and MAGE (Li et al., 2024) (1000 texts each). We report AUROC, F1, accuracy, and FPR and compare it to the results on the same sample for Binoculars and FastDetectGPT.

### 3.3 Model Capability Scaling

To investigate the relationship between model capability and detection accuracy, we evaluated 9 models across three families on a shared subset of the RAID benchmark (400 texts, stratified across 8 domains): 5 OpenAI models (GPT-3.5 Turbo through GPT-5.4), 4 Anthropic models (Claude Haiku 4.5 through Claude Opus 4.6), and open-source models (Mistral 7B (Jiang et al., 2023), and Gemma (Mesnard et al., 2024)). All models received the same detection prompt (Figure 3).

### 3.4 AI Prevalence Estimation on arxiv Papers

**Data Collection** We collect 1,000 papers from arxiv for all 48 months of 2022, 2023, 2024, 2025. 2022 text is expected to act as a baseline with near 0% AI detection rate as it is "pre-chatGPT" era. while we measure for prevalence over time in other years. After scoring all the paper segments using Just Ask with Opus 4.6, We retrieve the papers metadata from sematic scholar and analyze correlations between AI-detection rate and variables like field, author seniority, and publication status. more details on Data collection can be found in the appendix

## 4 Results

### 4.1 Benchmark Performance

JUST ASK with Claude Opus 4.6 achieves near-perfect detection on the RAID benchmark: 0.999 AUROC across all 8 domains (Table 1). At $\tau$=0.87, the method achieves 0.0% FPR, with a best F1 of 0.934 at $\tau$=0.65. On additional benchmarks, we observe 1.000 AUROC on HC3, 0.988 on M4GT-Bench, and 0.979 on MAGE.

Table 1 compares JUST ASK against existing open source detectors. Among zero-shot meth-

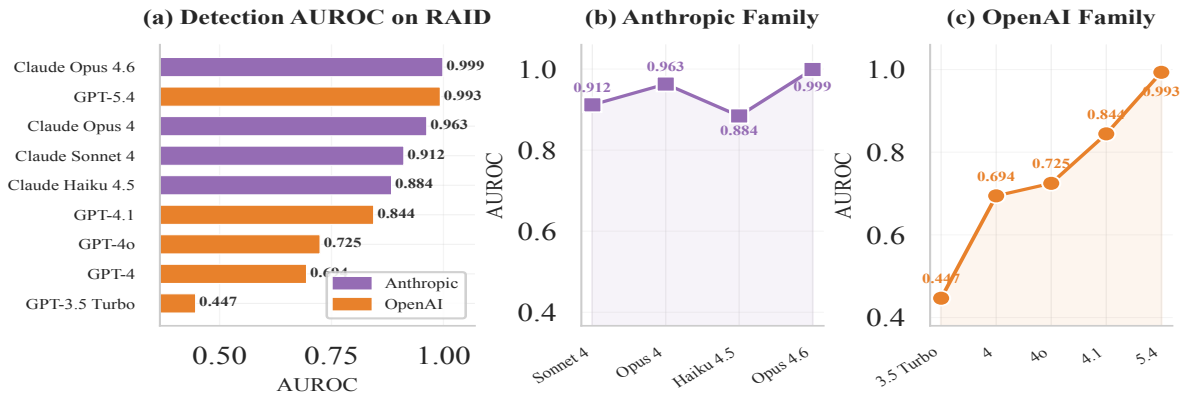


Figure 1: a. AI detection performance by model generation prompted with Just Ask b. AI detection performance on Anthropic models shows an increase in performance over time c. AI detection performance in OpenAI models shows an increase over time

| Bench. | Detector | AUROC | F1 | FPR |
|---|---|---|---|---|
| *RAID* | | | | |
| | Just Ask | **0.999** | 0.934 | **0.000** |
| | Binoculars | 0.982 | **0.971** | 0.020 |
| | F-DetectGPT | 0.418 | 0.667 | 0.996 |
| *HC3* | | | | |
| | Just Ask | **1.000** | **0.970** | **0.030** |
| | Binoculars | 0.998 | 0.960 | 0.084 |
| | F-DetectGPT | 0.261 | 0.664 | 0.998 |
| *M4GT-Bench* | | | | |
| | Just Ask (Opus 4.6) | **0.988** | **0.941** | 0.063 |
| | Binoculars | 0.914 | 0.911 | **0.020** |
| | F-DetectGPT | 0.455 | 0.667 | 1.000 |
| *MAGE* | | | | |
| | Just Ask | **0.979** | **0.933** | **0.068** |
| | Binoculars | 0.315 | 0.030 | 0.574 |
| | F-DetectGPT | 0.792 | 0.700 | 0.856 |
| *Overall* | | | | |
| | Just Ask | **0.986** | **0.946** | **0.061** |
| | Binoculars | 0.794 | 0.754 | 0.174 |
| | F-DetectGPT | 0.491 | 0.674 | 0.963 |

Table 1: Head-to-head comparison of zero-shot Open Source detectors, Binoculars with Falcon 7B, Fast-DetectGPT with gpt-neo-2.7B and Just Ask with Opus 4.6 on identical samples (500 human + 500 AI per benchmark). Best per benchmark in **bold**.

ods, JUST ASK outperforms Binoculars and Fast-DetectGPT on benchmarks. Furthermore, when tested on a synthetic positive control dataset produced by asking GPT-4o to "polish, expand, or rewrite" a chunk of paper text, it vastly outperforms both. It also matches the best commercial detectors on RAID (Grammarly and GPTZero, 0.999 AUROC) without requiring any training data.

### 4.2 Model Capability Scaling

Detection performance scales strongly with model capability (Table 3, Figure 1). Within the OpenAI family, AUROC increases monotonically from 0.447 (GPT-3.5 Turbo) to 0.993 (GPT-5.4). Anthropic models consistently outperform their OpenAI contemporaries, with Claude Opus 4.6 achieving the highest overall AUROC of 0.999. The correlation between model release date and detection AUROC is strong (Spearman $r > 0.8$, $p < 0.01$; Figure 1).

This scaling relationship suggests that AI text detection via direct querying is an emergent capability: weaker models (GPT-3.5 Turbo) perform at chance level, while frontier models achieve near-perfect discrimination.

### 4.3 Temporal Analysis

The temporal analysis of 48,000 arXiv papers reveals an exponential increase in AI-detected content (Figure 2). At $\tau$=0.87, the yearly detection rate grows from 0.1% in 2022 to 0.9% in 2023, 3.1% in 2024, and 9.2% in 2025 (Figure 2). The trend is robust across thresholds: at $\tau$=0.50, the rate rises from approximately 37% to 72% over the same period. The Spearman correlation between month index and detection rate exceeds 0.9 ($p = 0.000$). We will make the dataset public for future research and analysis.

Because our threshold yields 0.0% FPR on the RAID benchmark, these prevalence estimates are conservative lower bounds. The true rate of AI involvement may be higher, as lightly polished text is harder to detect than fully AI-generated text.

### 4.4 Metadata Analysis

We analyze correlations between AI detection rates and paper-level metadata for the 48,000-paper temporal corpus. Metadata is retrieved from the Se-

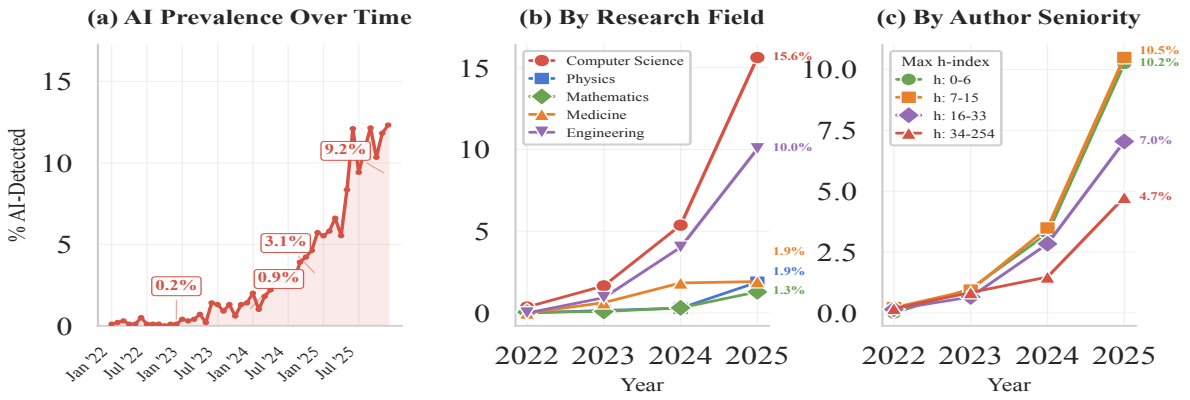


Figure 2: a. AI detection rate by month from Jan 2022 when to December 2025 (2022 rate of 0.1% serves as our baseline FPR on arxiv data since chatGPT launched in November 2022) b. AI detection rate by paper field over time shows Computer Science and Engineering with higher AI utilization and adoption speeds vs other fields c. AI detection rate vs max-h index of authors shows higher h-index authors utilize AI less than lower h-index authors

mantic Scholar API.

**Author Seniority.** The strongest correlate of AI detection is author seniority. The maximum $h$-index among a paper's authors correlates negatively with AI probability (Spearman $r{=}{-}0.222$, $p < 0.001$). When stratified by $h$-index quartile over time (Figure 2), low-$h$ and mid-low-$h$ groups show the steepest increase in 2025 (approximately 10%), while the high-$h$ group remains lowest (approximately 4.7%). First-author $h$-index ($r{=}{-}0.144$) and last-author $h$-index ($r{=}{-}0.164$) show similar but weaker correlations.

**Research Field.** AI detection rates vary sharply by field (Figure 2). Computer Science leads at 15.5% in 2025, followed by Engineering at approximately 10%. Physics, Mathematics, and Medicine remain below 3%. All fields show upward trends, but the rate of increase differs significantly.

**Publication Status.** Published (peer-reviewed) papers show slightly lower AI detection rates than preprint-only papers (Figure 7), suggesting that the peer review process may filter some AI-generated content or that authors submitting to venues exercise more caution. Conference papers and journal articles show similar rates (Figure 2).

**Cost Comparison.** JUST ASK with Claude Opus 4.6 costs approximately \$0.0034 per text, making it substantially cheaper than commercial detectors at scale. GPTZero charges \$0.015–\$0.025 per 256 word text depending on volume, making JUST ASK 78-86% cheaper. At the scale of our temporal analysis (48,000 text segments), the total cost was approximately \$168.

## 5 Discussion and Conclusion

We have presented JUST ASK, an effective zero-shot method for detecting AI-generated text that achieves SOTA performance matching finetuned commercial detectors without requiring any training data. we show that detection ability scales with model capability, suggesting that as LLMs improve, so too will their ability to detect AI-generated text.

Future work should investigate the method's robustness to adversarial attacks, extend the analysis to additional domains and languages, and explore whether "just ask" can detect more subtle forms of AI involvement such as AI-polished human writing.

## 6 Limitations

Our work has several important limitations. First, JUST ASK is black-box in nature: although we prompted the model to share the features that influenced its decision and show results in appendix, we do not have access to model internals and cannot provide mechanistic, verifiable explanations for why particular texts receive high or low scores, we can however see a trend towards increased performance with model generation. Second, For older datasets we evaluate only on the non-adversarial subset of RAID and do not claim robustness to deliberate evasion attempts such as paraphrasing attacks or adversarial perturbations. Third, our prevalence estimates are based on 256-word chunks from the middle third of papers, which may not represent the full document and could be systematically more

or less prone to authors using AI-assistance. Fourth, the method requires API access to a frontier LLM, creating a dependency on closed-source commercial providers as no available open source model has yet to match the AI detection performance of closed source ones. Fifth, while we validate against RAID, our evaluation uses a subset of 1,570 texts rather than the full 672K hidden test set which we couldn't test due to the prohibitive cost of doing so. As such, comparisons with leaderboard results should be interpreted with this caveat.

## References


George Adam, Alexander Cui, Edwin Thomas, E. R. Napier, Nazar Shmatko, Jacob Schnell, Jacob-Junqi Tian, Alekhya Dronavalli, Edward Tian, and Dongwon Lee. 2026. Gptzero: Robust detection of llm-generated texts. *ArXiv*, abs/2602.13042.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2023. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *ArXiv*, abs/2310.05130.

Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Christopher Callison-Burch. 2024. Raid: A shared benchmark for robust evaluation of machine-generated text detectors. *ArXiv*, abs/2405.07940.

Andrew Gray. 2024. Chatgpt "contamination": estimating the prevalence of llms in the scholarly literature. *ArXiv*, abs/2403.16887.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: Zero-shot detection of machine-generated text. In *International Conference on Machine Learning*.

Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2023. Mgtbench: Benchmarking machine-generated text detection. *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *ArXiv*, abs/2310.06825.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *International Conference on Machine Learning*.

Dmitry Kobak, Rita Gonz'alez-M'arquez, Emőke ágnes Horvát, and Jan Lause. 2024. Delving into llm-assisted writing in biomedical publications through excess vocabulary. *Science Advances*, 11.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2024. MAGE: Machine-generated text detection in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 36–53.

Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, Daniel A. McFarland, and James Y. Zou. 2024a. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on ai conference peer reviews. In *International Conference on Machine Learning*.

Weixin Liang, Yaohui Zhang, Mihai-Eugen Codreanu, Jiayu Wang, Hancheng Cao, and James Zou. 2025. The widespread adoption of large language model-assisted writing across society. *Patterns*, 6.

Weixin Liang, Yaohui Zhang, Zhengxuan Wu, Haley Lepp, Wenlong Ji, Xuandong Zhao, Hancheng Cao, Sheng Liu, Siyu He, Zhi Huang, Diyi Yang, Christopher Potts, Christopher D. Manning, and James Y. Zou. 2024b. Mapping the increasing use of llms in scientific papers. *ArXiv*, abs/2404.01268.

Gemma Team Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, L. Sifre, Morgane Rivière, Mihir Kale, J Christopher Love, Pouya Dehghani Tafti, L'eonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Am'elie H'eliou, Andrea Tacchetti, and 88 others. 2024. Gemma: Open models based on gemini research and technology. *ArXiv*, abs/2403.08295.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*.

Vishisht Srihari Rao, Aounon Kumar, Hima Lakkaraju, and Nihar B. Shah. 2025. Detecting llm-generated peer reviews. *PLOS One*, 20:e0331871 – e0331871.

Giuseppe Russo, Manoel Horta Ribeiro, Tim R. Davidson, Veniamin Veselovsky, and Robert West. 2024. The ai review lottery: Widespread ai-assisted peer reviews boost paper scores and acceptance rates. *Proceedings of the ACM on Human-Computer Interaction*, 9:1 – 28.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2023. Can AI-generated text be reliably detected? *Computing Research Repository*, arXiv:2303.11156.

Zhenpeng Su, Xing Wu, Wei Zhou, Guangyuan Ma, and Song Hu. 2023. Hc3 plus: A semantic-invariant human chatgpt comparison corpus. *ArXiv*, abs/2309.02731.

Vivek Kumar Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2023. Ghostbuster: Detecting text ghostwritten by large language models. In *North American Chapter of the Association for Computational Linguistics*.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. M4GT-bench: Evaluation benchmark for black-box machine-generated text detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 3964–3992.

**Algorithm:** JUST ASK Detection

**Require:** Target text $x$, frontier LLM $\mathcal{M}$, threshold $\tau$
**Ensure:** Detection label $\hat{y} \in \{0, 1\}$

1: Construct prompt $p$ from system message $s$ and user message $u(x)$
2: $r \leftarrow \mathcal{M}(s,\ u(x))$ ▷ Query LLM
3: $\hat{p} \leftarrow r.\texttt{probability_ai}$ ▷ Parse score
4: **return** $\hat{y} = \mathbb{1}[\hat{p} \geq \tau]$

**System message** $s$**:**

*You are an AI text detection system. You are well-calibrated: when you say 70% probability, you are correct about 70% of the time. A probability of 0.5 means you genuinely cannot tell. Do not default to extreme values unless you are very confident. Use the FULL range of probabilities: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. Avoid clustering your answers at 0.1 or 0.9.*

**User message** $u(x)$**:**

*Rate the probability (0.00 to 1.00) that the following text was generated by an AI language model. Consider specific evidence for and against. Use fine-grained probabilities, not just high/low.*

*Respond with ONLY:* `{"probability_ai": 0.XX}`

*Text:* $\langle x \rangle$

Figure 3: The JUST ASK method and prompt template.

## A Method Details

Figure 3 shows the complete JUST ASK algorithm and prompt template. The system message instructs the model to behave as a well-calibrated detector, emphasizing use of the full probability range. The user message presents the target text and requests a structured JSON response containing a single probability estimate.

## B Threshold Analysis

Threshold selection is critical for prevalence estimation. Table 2 shows detection performance at multiple thresholds on the RAID benchmark. At $\tau$=0.65, F1 is maximized at 0.991, but the false positive rate of 3.5% is unacceptable for prevalence measurement in low-base-rate settings. At $\tau$=0.87, FPR reaches 0.0%. We select $\tau$=0.87 as our operating threshold, accepting a modest reduction in recall to eliminate false positives entirely on this benchmark.

Figure 4 confirms that the temporal trend holds across thresholds: the year-over-year increase in AI-detected content is visible at every threshold tested.

| $\tau$ | FPR | TPR | F1 | Acc |
|---|---|---|---|---|
| 0.50 | 7.2% | 99.6% | 0.964 | 96.2% |
| 0.65 | 3.5% | 99.1% | 0.991 | 99.0% |
| 0.70 | 0.9% | 99.0% | 0.991 | 99.0% |
| 0.87 | 0.0% | 97.2% | 0.986 | 98.6% |

Table 2: Detection performance at multiple thresholds on the RAID benchmark (1,570 texts, 8 domains).

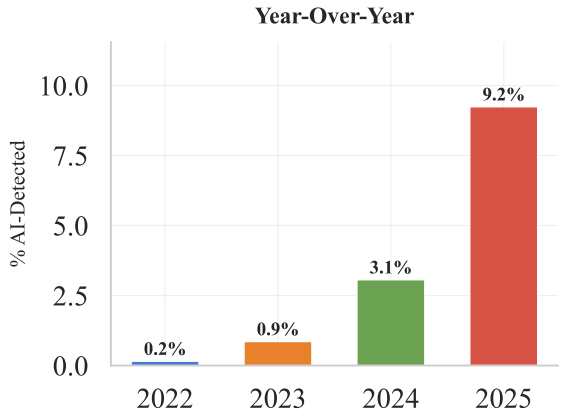


Figure 4: Year-over-year AI detection rates on arXiv at threshold $\tau$=0.87. The rate increases approximately threefold each year, from 0.1% in 2022 to 9.2% in 2025.

## C Model Capability Scaling

Table 3 reports per-model detection performance on the RAID benchmark for all 9 models evaluated. Within both the Anthropic and OpenAI families, AUROC increases monotonically with model generation. The strongest model (Claude Opus 4.6, AUROC 0.999) achieves near-perfect discrimination, while the weakest (GPT-3.5 Turbo, AUROC 0.447) performs below chance.

## D Positive Control: AI-Rewritten Text

To test whether detectors can identify AI-modified (as opposed to fully AI-generated) text, we constructed a positive control dataset. We sampled 300 human-written text chunks from pre-2022 arXiv papers (guaranteed pre-LLM) and had GPT-4o rewrite each chunk in three modes of increasing AI involvement:

- **Polish:** Light editing for clarity and flow, preserving the original structure and arguments. This represents the most common real-world use case of LLM writing assistance.

| Model | Release | AUROC | F1 |
|---|---|---|---|
| *Anthropic* | | | |
| Claude Sonnet 4 | 2025-05 | 0.912 | 0.945 |
| Claude Opus 4 | 2025-05 | 0.963 | 0.933 |
| Claude Haiku 4.5 | 2025-10 | 0.884 | 0.863 |
| Claude Opus 4.6 | 2026-02 | **0.999** | **0.967** |
| *OpenAI* | | | |
| GPT-3.5 Turbo | 2023-03 | 0.447 | 0.430 |
| GPT-4 | 2023-06 | 0.694 | 0.337 |
| GPT-4o | 2024-05 | 0.725 | 0.727 |
| GPT-4.1 | 2025-04 | 0.844 | 0.735 |
| GPT-5.4 | 2026-03 | 0.993 | 0.853 |

Table 3: Detection performance of JUST ASK across model families on the RAID benchmark. Best results in **bold**.

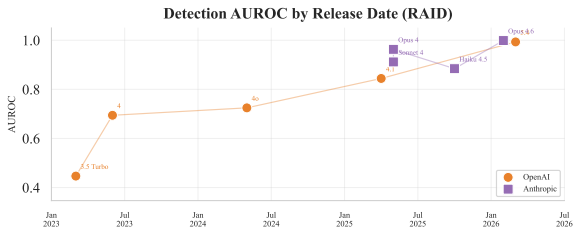


Figure 5: Detection AUROC by model release date. More capable models achieve higher detection accuracy across all three families (Spearman $r > 0.8$, $p < 0.01$).

- **Expand:** Elaborating on the original content, adding transitions and detail while retaining the core ideas.
- **Full rewrite:** Completely rewriting the passage from scratch on the same topic, producing essentially AI-generated text.

We scored all texts (300 originals + 900 rewrites) with Just Ask (Claude Opus 4.6), Binoculars (Falcon-7B), and Fast-DetectGPT (GPT-Neo-2.7B). Table 4 reports detection rates at each detector's default threshold.

Binoculars failed to detect any polished text (0.0%), and detects only 3.4% of fully rewritten text. This is consistent with known limitations of perplexity-based methods on text that retains human-like statistical properties. Just Ask with Claude Opus 4.6, by contrast, detects 25% of polished text and 78% of fully rewritten text at $\tau$=0.87, demonstrating substantially greater sensitivity to AI involvement across all rewrite modes.

This result is practically significant: in real-world academic writing, light polishing (grammar correction, sentence restructuring, clarity improvements) is far more common than full AI generation. A detector that cannot identify polished text will systematically underestimate AI prevalence. The ability of JUST ASK to detect even lightly polished text suggests that our prevalence estimates, while conservative, capture a broader range of AI involvement than perplexity-based methods could.

| Mode | Just Ask | Bino. | F-DGPT |
|---|---|---|---|
| Human original | 0.2% | 1.0% | 12.0% |
| Polish | 25.0% | 0.0% | 0.0% |
| Expand | 73.0% | 6.7% | 30.0% |
| Full rewrite | 78.0% | 3.4% | 28.0% |

Table 4: Detection rate on AI-rewritten arXiv text by rewrite mode. Just Ask uses $\tau$=0.87; Binoculars uses its default threshold of 0.9015. Binoculars fails entirely on polished text (0.0% detection), while Just Ask detects 25% of polished texts.

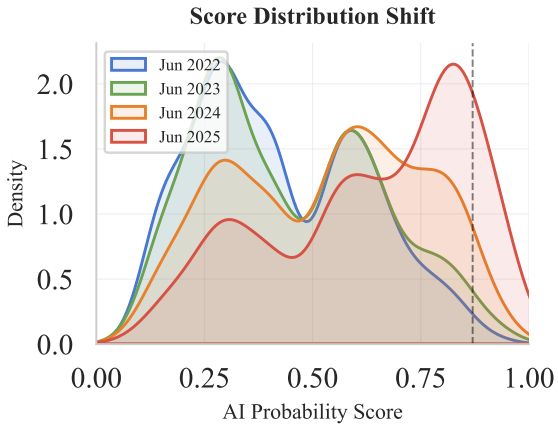


Figure 6: Score distribution shift across time. KDE plots of AI probability scores for arXiv papers sampled from June of each year show a clear rightward shift.

## E Additional Prevalence Analyses

Figure 6 shows the shift in AI probability score distributions over time. The rightward movement of the distribution from 2022 to 2025 reflects the growing fraction of texts receiving high AI probability scores.

Figure 7 compares AI detection rates for published (peer-reviewed) versus preprint-only papers. Both categories show upward trends, with preprints consistently exhibiting slightly higher detection rates, suggesting that the peer review process may filter some AI-generated content or that authors exercise more caution when targeting venues.

Figure 8 shows AI detection rates by team size. larger teams exhibit the highest detection rate.

## F Data Collection

**Temporal Corpus (arXiv).** We constructed a longitudinal corpus of 48,000 arXiv papers span-

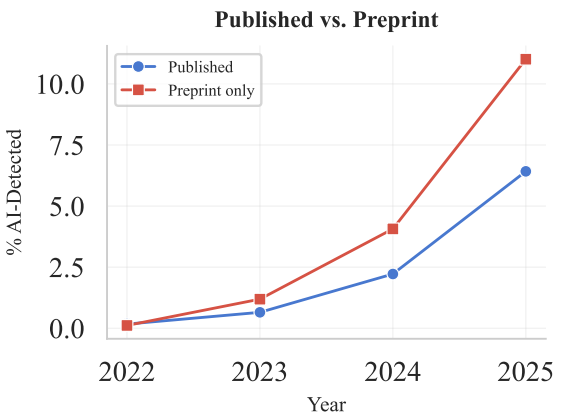


Figure 7: AI detection rate for published (peer-reviewed) versus preprint-only papers over time.

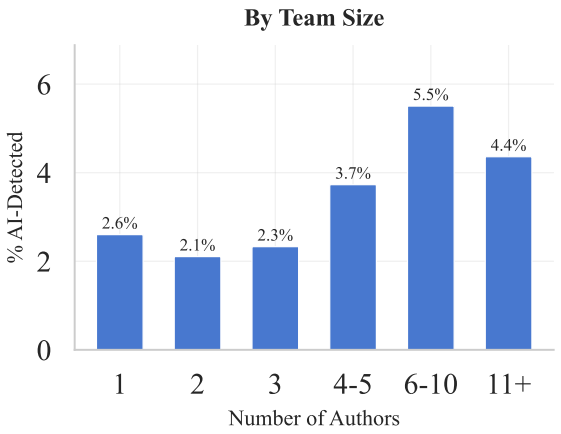


Figure 8: AI detection rate by number of authors. larger teams show higher AI generation assistance

ning January 2022 to December 2025, sampling 1,000 papers per month. For each paper, we stripped LaTeX markup, and removed the abstract, references, acknowledgments, appendix, figures, tables, and equations, then extracted a 256-word chunk from the middle third of the remaining body text, finding clean sentence boundaries at both ends. We then scored this chunk using "Just Ask".

**Metadata.** For the temporal corpus, we retrieve metadata from the Semantic Scholar API, including citation count, author $h$-indices, field of study, and publication venue. This enabled analysis of how AI adoption may vary with things like author seniority, research field, team size, and publication status.

## G Prompt Ablation Study

To assess the sensitivity of JUST ASK to prompt design, we evaluate eight prompt variants on the same 400-text RAID subset using Claude Opus 4.6.

| Variant | AUROC | F1 |
|---|---|---|
| Calibration + range + JSON | 0.999 | 0.966 |
| No system prompt | 0.999 | 0.975 |
| Calibration, no range instr. | 0.999 | 0.968 |
| Ask for P(human) instead | 0.997 | 0.980 |
| Minimal system prompt | 0.996 | 0.956 |
| Bare minimum, no instr. | 0.993 | 0.973 |
| Yes/no instead of prob. | 0.993 | 0.992 |
| Full + feature attribution | **1.000** | 0.983 |

Table 5: Prompt ablation results on 400-text RAID subset. AUROC ranges from 0.993 to 1.000 across all variants.

Each variant isolates a specific component: calibration instructions, range guidance, output format, or framing direction. The full text of each variant is shown in Table 7.

Table 5 reports AUROC and F1 for each variant. Detection is remarkably robust: AUROC ranges from 0.993 (minimal and binary variants) to 1.000 (features variant). Pairwise bootstrap comparisons show that only removing calibration ($p < 0.05$) or using the minimal prompt ($p < 0.05$) causes a statistically significant decrease relative to the full prompt—and even these decreases are small in absolute terms (0.003–0.006 AUROC).

The binary variant (yes/no output) achieves the highest F1 (0.992) but sacrifices the continuous probability scores needed for threshold tuning and prevalence correction. The reversed variant (asking for probability of human authorship) performs comparably to the standard framing, suggesting the model's detection ability is not sensitive to how the question is framed and is mainly a function of model capability.

## H Models Remembering Data

to rule out the possibility that larger models have simply seen and remembered all the human data in the benchmarks as well as in published research papers during their training or pre-training process, we re-ran the positive control experiment described in section D of the appendix with papers that are from June 2025 and after only which is after the knowledge cutoff of any of the used models eliminating this concern. The performance of the models did not degrade showing that they are not merely remembering the human input data from their training corpus.

## I Feature Attribution Analysis

When prompted to explain its detection decisions (the "features" variant), Claude Opus 4.6 returns a

| **Feature** | $N$ | $\mathbf{W_{hum.}}$ | $\mathbf{W_{AI}}$ |
|---|---|---|---|
| Structure | 88 | −0.21 | +0.56 |
| Opening phrase | 21 | +0.00 | +0.86 |
| Repetitive phrasing | 20 | −0.05 | +0.64 |
| Lexical diversity | 21 | −0.29 | +0.45 |
| Hedging language | 56 | −0.16 | +0.35 |
| Sentence structure | 24 | −0.28 | +0.53 |
| Abrupt ending | 16 | −0.45 | +0.46 |
| Writing style | 48 | −0.34 | +0.48 |
| Emotional authent. | 31 | −0.52 | +0.60 |
| Informal tone | 31 | −0.54 | +0.00 |
| Specific details | 40 | −0.47 | +0.00 |
| Direct quotes | 17 | −0.48 | +0.00 |

Table 6: Top features cited in detection explanations ($n$=250 texts). $W_{hum.}$ and $W_{AI}$ are mean self-reported weights when the true label is human or AI.

list of features with self-reported weights indicating how much each feature influenced its prediction. Table 6 summarizes the most frequently cited features across 250 texts with successfully parsed feature attributions.

The most discriminative feature is *structure*: when scoring AI-generated text, the model assigns it a strong positive weight (+0.56), but when scoring human text, it assigns a negative weight (−0.21). *Opening phrase* is the strongest single indicator of AI authorship (+0.86 for AI texts), likely reflecting the formulaic introductions common in LLM output. Conversely, *informal tone*, *specific details*, and *direct quotes* are cited exclusively as evidence of human authorship, consistent with the intuition that LLMs produce more generic, formal text.

The model cited 872 unique feature names across 250 texts (mean 8.4 features per text), but only one feature (*formatting*) appeared consistently across all 8 RAID domains. This suggests that the model adapts its detection rationale to the domain rather than relying on a fixed set of indicators.

| Variant | System Prompt | User Prompt |
|---|---|---|
| Full prompt | You are an AI text detection system. You are well-calibrated: when you say 70% probability, you are correct about 70% of the time. A probability of 0.5 means you genuinely cannot tell. Do not default to extreme values unless you are very confident. Use the FULL range of probabilities: 0.1, 0.2, ..., 0.9. Avoid clustering your answers at 0.1 or 0.9. | Rate the probability (0.00 to 1.00) that the following text was generated by an AI language model. Consider specific evidence for and against. Use fine-grained probabilities, not just high/low. Respond with ONLY: `{"probability_ai": 0.XX}` Text: $\langle x \rangle$ |
| No system | *(none)* | *(same as full prompt)* |
| No calibr. | You are an AI text detection system. | *(same as full prompt)* |
| No range | *(calibration only, no range/clustering instructions)* | *(same as full prompt)* |
| Minimal | *(none)* | Is this text AI-generated? Respond with ONLY: `{"probability_ai": 0.XX}` Text: $\langle x \rangle$ |
| Binary | You are an AI text detection system. | Is the following text generated by an AI language model? Respond with ONLY: `{"is_ai": true}` or `{"is_ai": false}` Text: $\langle x \rangle$ |
| Reversed | *(same as full prompt)* | Rate the probability (0.00 to 1.00) that the following text was written by a human (not AI-generated). Respond with ONLY: `{"probability_human": 0.XX}` Text: $\langle x \rangle$ |
| Features | *(same as full prompt)* | Analyze the text and return a JSON object with "probability_ai" (0.00–1.00) and "features": a list with name, value, weight ($-1$ to +1), and confidence (0–1). Respond with ONLY valid JSON. Text: $\langle x \rangle$ |

Table 7: Full prompt text for each ablation variant. Variants marked *(same as full prompt)* reuse that column from the full prompt row.