The Right Amount of Wrong: Error Injection for AI Oversight

Kacey Fang[1], Greg Powell[2], Jeffery Painter[2], Andrew Bate[2,3], and Michael Lingzhi Li[4]

[1]Harvard Medical School; [2]GSK; [3]London School of Hygiene & Tropical Medicine; [4]Harvard Business School

Corresponding author: Michael Lingzhi Li, mili@hbs.edu

August 5, 2026

Abstract

Organizations increasingly use AI systems to generate draft outputs that are reviewed by humans before downstream use, particularly in high-stakes settings such as healthcare. As AI systems become more accurate, however, their errors become increasingly rare, making reviewers less vigilant while simultaneously making reviewer performance difficult for organizations to observe. We study whether intentionally inserting realistic errors into selected AI outputs before review, a practice we refer to as controlled error injection, can improve human oversight while creating observable opportunities to measure reviewer performance. We investigate this question through a preregistered, double-blind, 12-week randomized experiment in AI-assisted pharmacovigilance review. Seventeen Doctor of Pharmacy (PharmD) reviewers evaluated AI-generated summaries of adverse event reports under a control condition and three levels of error injection, completing 10,088 reviews in a within-subject Williams Latin Square design. We find that the effect of controlled error injection is nonmonotone. Relative to the control condition, low-frequency error injection reduced true-error detection by 4 percentage points (a 17% decrease), whereas high-frequency error injection increased true-error detection by 7 percentage points (a 29% increase). False-positive detection did not differ significantly across conditions. Higher levels of error injection also increased review time and reduced trust in the AI system. Finally, reviewers who identified more injected errors were also more likely to identify naturally occurring AI errors. These findings suggest that controlled error injection can improve oversight in AI-assisted workflows while simultaneously providing organizations with a practical signal of reviewer vigilance. At the same time, the effectiveness of the intervention depends critically on the frequency of injected errors. Low levels of error injection reduced true-error detection, whereas higher levels improved it, suggesting that organizations should carefully calibrate the prevalence of injected errors when deploying this intervention.

Keywords: Human-AI collaboration, Error Injection, Randomized Experiment

Ethics statement: This study was reviewed and approved by the Harvard University Area Institutional Review Board under protocol IRB25-1200. All participants provided informed consent before participating in the study.

AI disclosure: LLMs were used as part of the experimental intervention to generate structured summaries of adverse event reports and to generate and validate intentionally injected errors, as described in Sections 5.2 and 5.3. The authors reviewed and are responsible for all manuscript content.

1 Introduction

Organizations increasingly rely on AI-assisted workflows to support high-stakes decisions. In these workflows, an AI system generates a draft output that is subsequently reviewed by a human expert before being used in downstream decision-making. Examples include physicians reviewing AI-generated medical documentation, content moderators reviewing AI recommendations, and pharmacovigilance specialists reviewing AI-generated case summaries [Agarwal et al., 2023, Brynjolfsson et al., 2025]. These workflows are attractive because they combine the speed and scalability of AI systems with human oversight.

Whether this workflow succeeds depends critically on the reviewer identifying and appropriately acting on errors in the AI output. In practice, however, this oversight is difficult to sustain and difficult to observe. First, as AI systems improve, errors become increasingly rare and review becomes repetitive, particularly in high-volume environments. Reviewers may therefore begin approving outputs without carefully checking them. This concern is consistent with prior evidence on automation bias [Bainbridge, 1983, Parasuraman and Manzey, 2010, Goddard et al., 2012] and the broader literature showing that rare targets are frequently missed even by trained reviewers [Wolfe et al., 2005, 2007, 2013]. Second, organizations often cannot directly measure whether reviewers are meaningfully checking the AI output. One approach is to assign a second expert to independently audit reviews, but this is expensive and often infeasible at the scale that motivates AI adoption in the first place. Organizations can therefore observe throughput and turnaround time, but not whether human oversight is functioning as intended.

In this paper, we study controlled error injection as a way to address both challenges and improve human oversight in AI-assisted workflows. Under controlled error injection, the system inserts synthetic errors into selected AI outputs before they are reviewed. Because these errors are known to the organization, they provide review opportunities with observable ground truth. Organizations can measure whether reviewers identify incorrect outputs and provide feedback when errors are missed. At the same time, injected errors alter the review environment by changing how frequently reviewers encounter incorrect AI-generated outputs. This may influence how carefully reviewers scrutinize AI-generated summaries during routine review and may affect process efficiency.

Related interventions have been studied in rare event monitoring and visual search settings. In airport baggage screening, for example, threat image projection systems insert artificial threats into X-ray images for training and performance evaluation [Hofer and Schwaninger, 2007, Riz à Porta et al., 2022, Buser et al., 2025]. However, there is limited experimental evidence on whether such interventions improve detection performance, and the available findings are mixed [Schwark et al., 2012, Taylor et al., 2022]. Whether controlled error injection improves performance in AI-assisted review workflows therefore remains unclear ex ante.

We study this question in postmarket drug safety surveillance. After a drug is approved, pharmaceutical companies continue to monitor adverse event reports to identify potential safety concerns that may not have been observed during clinical trials. This process is operationally important because pharmaceutical companies review large volumes of adverse event reports, and missed safety signals can have substantial clinical consequences. Given the scale of these reviews, pharmaceutical companies are increasingly exploring the use of generative artificial intelligence (AI) tools to generate structured summaries of adverse event reports for expert review [e.g. Hakim et al., 2025]. In a typical workflow, the AI generates a structured summary of an adverse event report, and

the reviewer verifies whether the extracted clinical information is correct before the report proceeds downstream.

To study the effect of controlled error injection in this setting, we conduct a preregistered, double-blind, longitudinal experiment in collaboration with a large pharmaceutical company. Participants are Doctor of Pharmacy (PharmD) students at the University of North Carolina who perform pharmacovigilance review tasks for twelve weeks. Each participant reviews 50 AI-generated summaries per week. Each summary contains eighteen structured clinical variables identified as clinically important in pharmacovigilance review [Kara et al., 2023]. During review, the system injects synthetic errors into selected summaries before they are shown to the reviewer. The experiment varies the frequency of injected errors across four conditions: no injected errors and injected errors generated according to Poisson distributions with mean rates of 1/3, 1, and 3 errors per summary. Participants remain in each condition for three weeks before rotating to the next condition according to a Williams Latin square crossover design, allowing within-participant comparisons across intervention levels while balancing first-order carryover effects.

We find that controlled error injection has nonmonotonic effects on reviewer vigilance. Low levels of injection reduce reviewers' ability to identify true AI errors relative to the no-injection condition, whereas high levels of injection increase detection of both injected errors and true AI errors. These gains do not appear to come solely from reviewers spending more time on each summary. Although high injection increases review time and reduces trust in the AI system, reviewers in the high-injection condition also identify more true errors per unit time than reviewers in the control condition, though this efficiency difference is not statistically significant after adjustment for multiple comparisons. Together, these results show that error injection can improve oversight in AI-assisted workflows, but only when injected errors are sufficiently prevalent to change reviewer behavior.

Overall, this paper makes three contributions. First, it introduces controlled error injection as an operational mechanism for sustaining and measuring human oversight in AI-assisted workflows. Organizations increasingly rely on human review as a safeguard against AI errors, but reviewer performance is difficult to observe because naturally occurring AI errors are sparse, heterogeneous, and often unknown to the organization at the time of review [Langer et al., 2025]. Controlled error injection addresses this measurement problem by creating review opportunities with known ground truth, allowing organizations to observe whether reviewers identify incorrect AI outputs.

Second, we provide some of the first experimental evidence on the effectiveness of controlled error injection in an AI-assisted review workflow. Using a 12-week longitudinal experiment in pharmacovigilance review, we show that controlled error injection statistically significantly affects reviewers' ability to identify true AI-generated errors.

Third, we show that the same intervention can improve or worsen oversight depending on the prevalence of injected errors. Low levels of injection reduce detection of true AI-generated errors relative to the no-injection condition, whereas high levels increase detection of both injected errors and true AI-generated errors. The prevalence of injected errors therefore becomes an important design choice in AI-assisted workflows.

2 Literature Review

Our paper relates to the growing literature on human-AI collaboration, where an algorithm provides predictions, recommendations, or generated content that a human decision-maker uses in a downstream task [Dell'Acqua et al., 2023, Grand-Clément and Pauphilet, 2026, Vaccaro et al., 2024]. Much of this literature studies settings in which algorithms provide recommendations and humans retain final decision authority [Kesavan and Kushwaha, 2020, Kwon et al., 2022, Boussioux et al., 2024, Balakrishnan et al., 2026]. Humans may remain in the loop because algorithms are imperfect, because humans possess information unavailable to the algorithm, or because final authority is retained for reasons of accountability and professional judgment [Bainbridge, 1983, Lebovitz et al., 2022, Sun et al., 2022]. Our work contributes to this literature by proposing and evaluating controlled error injection as a mechanism for sustaining human oversight in AI-assisted review workflows.

Our work is also related to research on automation bias and vigilance in rare-event monitoring. Prior work shows that human operators often rely too heavily on automated decision aids and consequently miss system errors [Parasuraman and Manzey, 2010, Goddard et al., 2012]. Related work on vigilance shows that monitoring performance deteriorates when tasks are repetitive and require sustained attention [Mackworth, 1948, Warm et al., 2008]. A broader experimental literature finds that rare targets are frequently missed and that detection performance deteriorates when target prevalence is low [Wolfe et al., 2005, 2007, 2013]. The closest operational analogue to our intervention appears in airport baggage screening, where threat image projection systems insert artificial threats into X-ray images for training and performance evaluation [Hofer and Schwaninger, 2007, Riz à Porta et al., 2022, Buser et al., 2025]. Experimental evidence on whether such interventions improve detection performance remains mixed [Schwark et al., 2012, Taylor et al., 2022]. More recently, Yin et al. [2026] theoretically shows that controlled error injection may provide a solution to the problem of incentivizing human oversight as AI systems become increasingly reliable. Our paper complements this contemporary theoretical work by providing causal experimental evidence from a real AI-assisted review workflow. We show that the effectiveness of controlled error injection depends critically on the prevalence of injected errors, with low-frequency injection degrading human oversight and high-frequency injection substantially improving it.

Finally, our paper contributes to healthcare operations and pharmacovigilance. Postmarket drug safety surveillance requires firms and regulators to monitor adverse event reports after approval because important safety issues may emerge only after drugs are used at scale in routine practice [Downing et al., 2017]. Pharmacovigilance review is labor intensive because adverse event reports are numerous, heterogeneous, and clinically complex. Prior work has developed automated methods to prioritize reports with high clinical utility [Kara et al., 2023], while more recent work has explored large language model applications in pharmacovigilance and highlighted the need for safeguards in safety critical workflows [Hakim et al., 2025]. We study one such safeguard. Rather than evaluating whether AI can generate useful pharmacovigilance summaries, we examine how organizations can design review processes that maintain effective human oversight after AI-generated summaries become the object of review.

3 Pharmacovigilance Review and Error Injection

Postmarket pharmacovigilance is an important component of the drug safety monitoring process. Although clinical trials evaluate the safety and efficacy of a drug before approval, these trials are conducted on relatively limited patient populations under controlled conditions. After approval, drugs are prescribed across broader and more heterogeneous patient populations, often over longer time horizons and in combination with other therapies. As a result, adverse drug reactions and safety concerns may emerge only after widespread clinical use [U.S. Food and Drug Administration, 2017, 2025].

A major input into postmarket pharmacovigilance is the adverse drug event case report. These reports may originate from physicians, patients, published case reports, manufacturers, or other reporting channels. Pharmacovigilance teams review these reports to identify clinically relevant safety information and determine whether the case should enter downstream safety monitoring and regulatory reporting processes [U.S. Food and Drug Administration, 2017, 2018, Kara et al., 2023]. In practice, this creates a substantial operational burden because reviewers must process large numbers of lengthy and heterogeneous reports containing unstructured clinical information [Kara et al., 2023, Dowdy et al., 2024].

Traditionally, pharmacovigilance reviewers read the adverse event report directly and manually prepare a structured case summary for downstream review. More recently, pharmaceutical companies have begun trialing generative AI systems to generate these structured summaries automatically. The reviewer therefore no longer prepares the summary from scratch. Instead, the reviewer compares the AI-generated summary against the underlying report, corrects any inaccurate fields, and submits the reviewed summary into the downstream pharmacovigilance process [Dowdy et al., 2024]. This type of AI-assisted review process is also now prevalent in a wide range of other operational settings, including content moderation, medical documentation, insurance claims review, and customer support workflows [Lykouris and Weng, 2025, Van Tiem et al., 2026].

The AI assistance fundamentally changes the operational role of the human. Under manual review, the reviewer constructs the output directly from the source material. Under AI-assisted review, the reviewer primarily monitors an already completed draft for errors. This shift can improve productivity, but it also creates an oversight problem. The reviewer must remain attentive even when most fields appear correct.

This oversight problem becomes more difficult as AI systems improve. When AI errors are frequent, reviewers receive regular reminders that the AI output must be verified carefully. As AI errors become rare, review instead becomes a low-prevalence detection task. Reviewers may process long sequences of apparently correct summaries, making errors less salient and reducing the perceived value of verifying each field against the underlying report. Prior work on automation bias and automation complacency documents similar patterns in monitoring and decision-support settings, where users under-monitor highly reliable automated systems and become less likely to detect infrequent failures [Parasuraman and Manzey, 2010, Goddard et al., 2012]. Related evidence from visual search shows that rare targets are detected substantially less often than common targets, even by trained reviewers [Wolfe et al., 2005, 2007, 2013]. As a result, organizations face a vigilance problem: maintaining careful human oversight becomes increasingly difficult as AI systems become more reliable. It can also become increasingly expensive. Recent work shows that, under accuracy-based compensation, the incentives required to sustain careful oversight grow without

bound as AI error rates approach zero [Bastani and Cachon, 2025, Yin et al., 2026].

The organization faces a second challenge: it often cannot observe whether oversight is functioning. Throughput, turnaround time, and the number of corrected fields are observable, but these measures do not reveal whether reviewers carefully verified the source report or simply accepted the AI-generated summary. Directly measuring missed AI errors would require an independent expert audit of every reviewed case, which is costly and difficult to scale. As a result, organizations face a measurement problem: they may observe that reviews are completed quickly without knowing whether the human oversight layer is detecting the errors it is intended to catch.

To address these two problems, we introduce controlled error injection. The basic idea is to intentionally insert realistic synthetic errors into selected AI-generated summaries before human review. By increasing the frequency with which reviewers encounter errors in AI-generated summaries, controlled error injection may change how reviewers approach subsequent reviews and help sustain reviewer vigilance. At the same time, because the injected errors have known ground truth, organizations can directly evaluate reviewer performance, provide immediate feedback when injected errors are missed, and tie incentives to review accuracy. Controlled error injection therefore aims to mitigate both the vigilance problem and the measurement problem in AI-assisted review.

4 Theory for Error Injection

Error injection can affect reviewers through two competing mechanisms. On one hand, encountering injected errors may recalibrate reviewers' beliefs about the reliability of AI-generated summaries. As errors become more frequent and salient, reviewers may become less willing to accept AI outputs at face value, increasing the perceived return to verifying the underlying source material. Under the same accuracy-based incentive, this shift may lead reviewers to devote greater attention to verification and improve detection of naturally occurring AI errors. On the other hand, injected errors also create additional discrepancies that reviewers must identify and resolve, consuming time and attention that could otherwise be devoted to naturally occurring errors.

The net effect therefore depends on the prevalence of injected errors. When injected errors are rare, they may increase review burden without meaningfully changing reviewers' beliefs about AI reliability. At sufficiently high levels, however, injected errors may make AI errors common enough for reviewers to update their perceived error prevalence and scrutinize AI-generated summaries more carefully. In this case, the increase in reviewer vigilance may outweigh the additional review burden. Error injection is therefore most likely to improve oversight when the injected-error rate is sufficiently high to recalibrate reviewers' perceived error prevalence, potentially approaching or exceeding the prevalence of naturally occurring AI errors. Appendix A presents a toy structural model that formalizes this intuition.

Hypothesis 1. Error injection has a nonmonotone effect on reviewers' detection of naturally occurring AI errors. Sufficiently frequent error injection improves detection, whereas sparse error injection can reduce detection.

If higher levels of error injection recalibrate reviewers' beliefs about AI reliability, reviewers should become less willing to accept AI-generated summaries at face value and devote greater effort to verification.

Hypothesis 1a. Sufficiently frequent error injection increases review effort.

Hypothesis 1b. Sufficiently frequent error injection reduces reviewers' trust in AI-generated summaries.

If reviewers respond by increasing scrutiny rather than indiscriminately questioning AI-generated summaries, improvements in true-error detection need not be accompanied by a corresponding increase in false-positive flagging.

Hypothesis 1c. Sufficiently frequent error injection does not produce a corresponding increase in false-positive flagging.

Controlled error injection also creates review opportunities with known ground truth. Naturally occurring AI errors are difficult for organizations to observe because determining whether a reviewer overlooked an error requires an independent audit. In contrast, organizations know exactly which errors were intentionally injected. If detection of injected errors and naturally occurring AI errors are both driven by reviewer vigilance, reviewers who identify more injected errors should also identify more naturally occurring AI errors. Appendix A formalizes this implication.

Hypothesis 2. Reviewer-level detection of injected errors is positively associated with reviewer-level detection of naturally occurring AI errors.

5 Experimental Design

To study the effectiveness of error injection, we collaborated with a large pharmaceutical company to develop a pharmacovigilance review platform that mimics the operational structure of real-world AI-assisted review. We preregistered the experimental design and statistical analysis plan before data collection. The preregistration is publicly available through the Open Science Framework Registries at `https://osf.io/mq7cv/overview`.

5.1 Participant Recruitment

Participants were PharmD students at the University of North Carolina at Chapel Hill (UNC) Eshelman School of Pharmacy. PharmD students were selected because their clinical and pharmaceutical training provided a relevant foundation for interpreting adverse drug event reports, and pharmacists commonly perform pharmacovigilance reviews in real-world settings. To be eligible, reviewers were required to be in good academic standing at the start of the study, as determined by the UNC Office of Experiential Education, to reduce the risk that study participation would interfere with students' ability to meet curricular requirements.

Participants were recruited through direct outreach to PharmD students at the University of North Carolina. Of the 31 students contacted, three did not meet the eligibility criterion of being in good academic standing. Twenty-two eligible participants consented to participate and completed the training period. Of these, three did not begin the experimental period because of changes in availability. One participant was excluded in accordance with the preregistered protocol after failing to complete more than 50 assigned reviews. A second participant was excluded because their review times were implausibly short relative to the task requirements, with a median review time of 38 seconds per article compared with approximately 300 seconds for the remaining participants.[1] The

[1]This exclusion was not prespecified in the preregistration. The results are materially unchanged when this participant is retained.

final analytic sample consisted of 17 participants. Appendix Figure 10 summarizes participant recruitment, study participation, and exclusions.

5.2 Experimental Materials and Review Platform

The experimental task required participants to review AI-generated summaries of adverse drug event reports. To construct the experimental materials, we first identified 1,085 publicly available reports through a PubMed search. The full search strategy is provided in Appendix G. We then excluded articles that did not describe at least one suspect drug, did not describe an adverse event, did not include an individual case report, were not written in English, or were not available in full text. After applying these criteria, 748 eligible reports remained. We selected 600 of these reports for inclusion in the experiment. GPT-4o mini was then used to generate a structured summary of each selected report. The complete summary-generation prompt is provided in Appendix H.

Each summary contained 18 clinical variables commonly used in pharmacovigilance review, including the suspect drug, medical history, and dosing regimen. The complete set of variables is presented in Table 1. These fields were selected based on prior work identifying the information most relevant to pharmacovigilance review [Kara et al., 2023]. To ensure consistency in both summary generation and review, we developed a standardized review manual defining each variable and providing examples of correct entries and common errors. The full manual is provided in Appendix B.

A team of four medical students, who did not participate in the experiment, were trained on the annotation manual and independently reviewed the summaries to identify errors in the AI-generated content. To ensure consistency in error labeling, all summaries underwent multi-reviewer assessment and disagreements were resolved through discussion according to predefined adjudication procedures. A true error was defined as a meaningful deviation from the source report that could affect interpretation of the case, including unsupported additions, incorrect extractions, misclassifications, or omissions of relevant information. Reviewers considered the context of the report when determining whether a discrepancy constituted an error. Minor technical discrepancies were not classified as errors when the intended meaning was clear and the inference was unambiguous from context. For example, an inferred route of administration was not considered an error when only one plausible route existed or when the route was otherwise clear from the report. Minor grammatical, stylistic, and formatting issues that did not alter the clinical meaning were also not classified as errors.

Participants completed reviews through an online, self-administered study platform. The platform was designed to approximate the review process used in real AI-assisted pharmacovigilance reviews. The review platform displayed the source report and the AI-generated summary side by side (Figure 1). For each summary, participants compared the structured variables with the underlying adverse drug event report and marked any variables they believed were incorrect, or marked that the summary contained no errors. Participants could highlight text and take notes while reviewing reports. Copying and pasting from the source report was disabled. To discourage superficial review, the platform displayed an attention-check prompt whenever a review was submitted in less than 30 seconds. Reviews were automatically saved and a prompt appeared after 90 seconds of inactivity.

Table 1: Variables extracted from adverse drug event reports

Variable	Definition
Age	Patient age, using the most specific information reported.
Gender	Patient gender as stated or clearly indicated.
Historical drug(s)	Medication(s) discontinued before the suspect-drug exposure period.
Medical history	Pre-existing conditions, comorbidities, and relevant substance-use history.
Suspect drug(s)	Drug(s) identified as the likely cause of the event, including interacting drugs when applicable.
Indication	Reported reason the suspect drug was prescribed or taken.
Dosing regimen	Reported dose, frequency, duration, and route of the suspect drug.
Start date of suspect drug	Date or relative timing of suspect-drug initiation.
Stop date of suspect drug	Date or relative timing of discontinuation, or indication that the drug was continued.
Disposition of drug	Action taken regarding the suspect drug, such as discontinuation, dose reduction, or continuation.
Concomitant medication(s)	Medication(s) taken during the same period as the suspect drug.
Event(s)	Reported adverse-event symptoms and/or diagnoses.
Event onset date	First reported date or relative timing of adverse-event onset.
Time to onset	Interval between suspect-drug initiation and event onset.
Treatment product(s)	Medication(s), procedures, or supportive interventions used to manage the event.
Outcome of adverse event	Reported clinical outcome, such as resolved, improved, worsened, fatal, or unknown.
Dechallenge	Response after suspect-drug discontinuation: positive, negative, or unknown.
Rechallenge	Response after systemic re-exposure: positive, negative, or unknown.

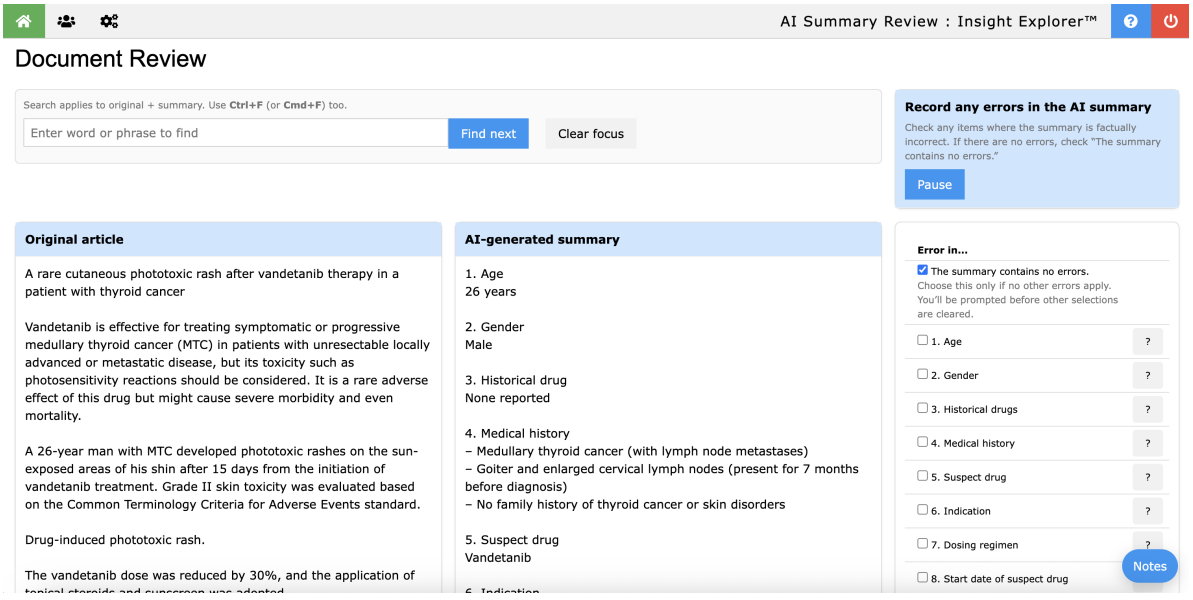

Figure 1: Review platform interface.

5.3 Controlled Error Injection

For each AI-generated summary, we implemented controlled error injection by introducing synthetic errors into the summary using an AI model. For each injected error, the model also generated

a brief explanation describing the error. If a participant missed an injected error, the platform displayed the corresponding feedback (Figure 2). Participants were informed that the feedback identified some, but not necessarily all, errors they had missed and that additional errors could be present. No feedback was provided for true AI-generated errors that were not intentionally injected or for variables reviewed correctly. This implementation is consistent with previous systems that introduce artificial threats to monitor and improve human performance [**?**].

The experimental manipulation was the rate of intentionally injected errors. For each adverse drug event report, we created four versions of the structured summary corresponding to the four experimental conditions: no-injection, low-injection, medium-injection, and high-injection. The no-injection version retained the original AI-generated summary without modification. The remaining three conditions introduced additional synthetic errors into the summary.

The high-injection version was generated first. For each summary, the number of injected errors was sampled from a Poisson distribution with a mean of 3 errors per summary. For summaries assigned one or more injected errors, GPT-5.1 was provided with the source report, the original structured summary, the standardized review manual, and examples of realistic failure modes. The model was instructed to identify pre-existing errors internally and then introduce the assigned number of new errors into distinct variables that it identified as otherwise correct. This approach reflects a realistic implementation setting in which the locations of naturally occurring AI errors would not be known before human review.

Injected errors were designed to resemble realistic AI summarization failures. Each injected error was required to be meaningful, directly verifiable from the source report, and representative of plausible mistakes that could occur during automated summarization. To verify that injected errors satisfied these criteria, each proposed error was independently evaluated three times by GPT-5.1 using the standardized review manual and the original source report. Errors classified as invalid in any evaluation were manually reviewed by the research team. If manual review determined that a proposed error did not represent a genuine deviation from the source report, the model generated a replacement error in the same variable and the validation procedure was repeated. Table 2 presents an illustrative example of an injected error, and the complete taxonomy is provided in Appendix C. The model was also instructed to preserve the wording of unmodified variables, avoid overwriting pre-existing errors whenever possible, and diversify both the variables modified and the categories of errors introduced. Appendix Figure 11 shows the distribution of naturally occurring true errors and injected errors across the 18 summary variables. Partly reflecting the design constraint that injected errors should not overwrite naturally occurring true errors, injected errors were relatively more concentrated in fields such as events, event outcome, and indication, whereas true errors were relatively more concentrated in fields such as historical drugs, medical history, time to onset, event onset date, and treatment products. Although injected errors differed from naturally occurring true errors in their variable-level distribution, the injected-error distributions were similar across the low-, medium-, and high-injection conditions, suggesting that comparisons across injected-error conditions were not driven by differences in where injected errors were placed.

The medium- and low-injection versions were then derived from the high-injection version by retaining subsets of the injected errors and reverting the remaining modified variables to their original values. The retained errors were randomly selected to approximate target mean injection rates of 1 error per summary in the medium-injection condition and 1/3 error per summary in the

low-injection condition. The resulting summaries therefore differed only in the number of injected errors presented during review. This design isolated the effect of injected-error prevalence while holding the underlying reports, AI-generated summaries, and error-generation process constant across conditions.

Table 2: Illustrative AI failure mode in adverse drug event report summarization

Error category	**Description**	**Illustrative example**
Incorrect field classification	The summary extracts information from the report but assigns it to the wrong field. For example, it may classify a historical drug, suspect drug, or treatment product as a concomitant medication, or classify an adverse event as medical history.	The patient received tacrolimus to treat the adverse event after the suspect drug was discontinued, but the summary lists tacrolimus as a concomitant medication.

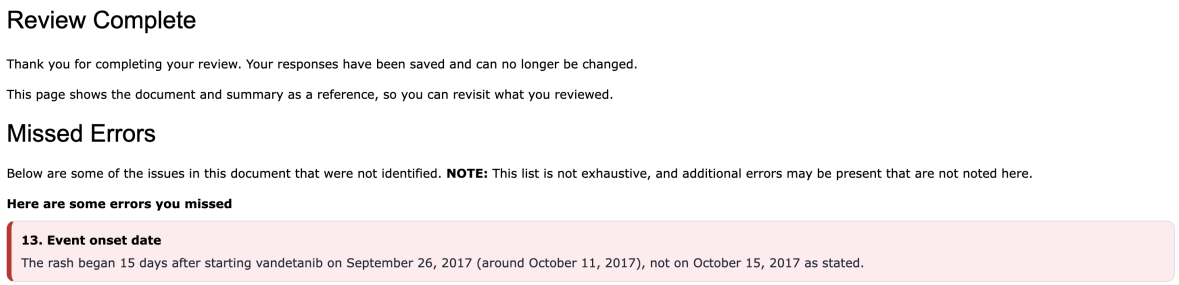

Figure 2: Example of automated feedback displayed after a participant failed to identify an intentionally injected error.

5.4 Experimental Procedure

The study consisted of a one-week training period followed by a 12-week experimental period. The experimental period used a preregistered, double-blind, within-subject crossover design. Participants remained in each condition for three weeks before rotating to the next condition, producing four three-week treatment periods. To control for period and carryover effects, treatment order was assigned using a Williams Latin square design [Williams, 1949]. The four treatment sequences are shown in Table 3.

Table 3: Williams Latin square treatment assignment

	Treatment period			
Sequence	Weeks 1–3	Weeks 4–6	Weeks 7–9	Weeks 10–12
1	Control	Low	High	Medium
2	Low	Medium	Control	High
3	Medium	High	Low	Control
4	High	Control	Medium	Low

This design provides two advantages. First, each participant experiences all four treatment conditions, allowing treatment effects to be estimated from within-participant comparisons rather than between-participant differences. Second, the Williams Latin square balances first-order

carryover effects by ensuring that each condition appears equally often before and after every other condition. As a result, treatment effects are not confounded with treatment order or immediate carryover from the preceding condition.

During the experimental period, each participant was asked to review 50 summaries per week, for a total of 600 summary-level reviews. Because each summary contained 18 structured clinical variables, the design generated up to 10,800 variable-level observations per participant. Participants were assigned to treatment sequences using covariate-constrained rerandomization with fixed sequence counts. Specifically, we generated 5,000 candidate allocations and selected the allocation that minimized the maximum absolute baseline imbalance t-statistic across prespecified covariates: year in the PharmD program, baseline trust in AI, attention-related cognitive errors, familiarity with adverse drug event reports, and familiarity with large language models. Baseline characteristics were similar across assigned sequences before and after participant dropout and exclusion (Table 4). Slightly unequal numbers of participants across treatment sequences were mitigated by the within-subject crossover design and by estimating treatment effects with reviewer and study-period fixed effects, cumulative review order, and reviewer-clustered standard errors.

Table 4: Baseline characteristics by randomized sequence

Characteristic	**Overall**	**S1**	**S2**	**S3**	**S4**	***p*-value**
Participants, n	17	4	5	5	3	
Age	24.88 (1.32)	24.50 (1.91)	25.40 (1.14)	25.00 (1.00)	24.33 (1.53)	0.686
Year in PharmD program	3.47 (0.87)	3.50 (0.58)	3.40 (1.14)	3.40 (0.55)	3.67 (1.53)	0.980
Familiarity with reviewing adverse drug event reports	2.76 (0.75)	3.25 (0.50)	2.40 (1.14)	2.80 (0.45)	2.67 (0.58)	0.441
Frequency of LLM use	3.65 (1.11)	3.75 (0.96)	4.00 (0.71)	3.60 (1.34)	3.00 (1.73)	0.711
Familiarity with LLMs	2.94 (0.66)	3.00 (0.00)	2.80 (0.45)	3.20 (0.45)	2.67 (1.53)	0.711
General trust in AI-system outputs	3.12 (0.93)	3.50 (1.00)	3.00 (1.00)	2.80 (0.84)	3.33 (1.15)	0.721
Trust in AI-generated adverse drug event summaries	3.00 (0.79)	3.00 (1.15)	3.40 (0.55)	2.80 (0.84)	2.67 (0.58)	0.587
Baseline attention-related cognitive errors	2.60 (0.49)	2.42 (0.24)	2.80 (0.65)	2.62 (0.65)	2.50 (0.08)	0.722

Note: Values are mean (standard deviation) unless otherwise indicated. p-values are from one-way analyses of variance across randomized sequences.

Participants were blinded to their assigned condition at any given time. Personnel interacting directly with participants were also blinded to treatment assignment. When participants rotated to a new condition, they were informed that they were using a new AI system. This procedure simulated a system restart and reduced the likelihood that participants would infer the underlying manipulation.

Before entering the experimental period, participants completed a one-week training program. Participants first watched an instructional video describing the review platform and task. They then completed 39 practice reviews using summaries for which the research team had previously identified the true errors. During training, participants received feedback on all missed errors and false-positive flags. Participants could also ask questions during designated office hours or by email and submit feedback through email or an anonymous form.

For each review, the platform recorded completion time and the variables marked as incorrect. These records were used to construct the primary behavioral outcomes. Sensitivity was defined as correctly identifying variables that contained a true non-injected error, and false positives were

defined as incorrectly flagging truly correct variables as erroneous. Review time was measured as the elapsed time between summary display and review submission.

Participants completed surveys at baseline, after the training period and before the first experimental week, and after each experimental week. The surveys measured subjective task load using the NASA Task Load Index (NASA-TLX) [Hart and Staveland, 1988] and trust in AI using Likert-scale items, including two substantive trust items and attention-check items. Surveys also included attention-check questions instructing participants to select a specified response. The complete survey instruments are provided in Appendix I.

To promote sustained participation, participants received both fixed compensation and a performance-based bonus. The fixed payment was $60 per two-week period, for a total of $390 over the 13-week study. Participants were also eligible to receive up to $90 in performance-based bonuses based on review accuracy, a weighted score that assigned 80% weight to sensitivity and 20% weight to specificity.

6 Results

6.1 True Error Detection

We first test Hypothesis 1 by examining whether controlled error injection changes reviewers' ability to identify naturally occurring AI errors. We estimate the following logistic regression model:

$$\operatorname{logit} \Pr(Y_{idvt} = 1) = \beta_0 + \mathbf{C}'_{idvt}\boldsymbol{\beta} + \theta \operatorname{Order}_{idt} + \gamma_t + \alpha_i + \delta_d + \lambda_v + \operatorname{Order}_{idt} \left(\mathbf{Z}'_i\boldsymbol{\eta}\right). \tag{1}$$

where Y_{idvt} is an indicator equal to 1 if reviewer i flagged a true error in variable v of document d during experimental period t. The vector $\mathbf{C}_{idvt}$ contains indicators for the low-, medium-, and high-injected-error conditions; the omitted reference category is the control condition. The terms γ_t, α_i, δ_d, and λ_v denote experimental-period, reviewer, document, and variable fixed effects, respectively. $\operatorname{Order}_{idt}$ denotes the cumulative order in which reviewer i reviewed document d during experimental period t, and captures systematic changes in reviewer behavior over the course of the experiment, such as learning or fatigue. The vector $\mathbf{Z}_i$ contains reviewer-specific baseline characteristics measured prior to the experiment, including familiarity with AI, trust in AI, and attention-related cognitive errors. Because these characteristics are time invariant, their main effects are absorbed by the reviewer fixed effects. The interaction term $\operatorname{Order}_{idt} \left(\mathbf{Z}'_i\boldsymbol{\eta}\right)$ allows the effect of review order to vary with reviewers' baseline characteristics, permitting reviewers with different baseline familiarity with AI, trust in AI, and attention-related cognitive errors to exhibit different learning or fatigue trajectories over the course of the experiment. Standard errors are clustered at the reviewer level.

Figure 3 presents the main results. Reviewers in the control condition identified approximately 24% of the true errors present in the summaries. This unadjusted detection rate fell to approximately 20% in the low-injected-error condition, remained similar at approximately 25% in the medium-injected-error condition, and increased to approximately 31% in the high-injected-error condition.

These results support Hypothesis 1. Relative to the control condition, reviewers in the high-injected-error condition had substantially higher odds of detecting a true error (OR, 1.46; 95% CI, 1.18–1.80; adjusted $p < 0.01$), whereas reviewers in the low-injected-error condition had significantly

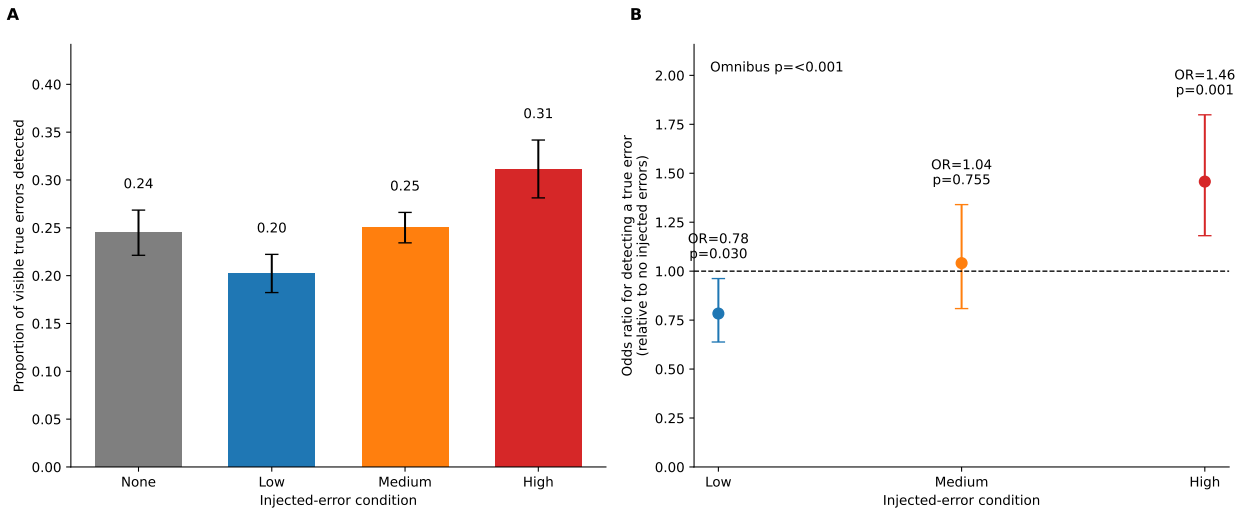

Figure 3: **True-error detection varies nonmonotonically with injected-error frequency.** Panel A reports raw true-error detection rates. Panel B reports adjusted odds ratios relative to the control condition. Error bars indicate 95% confidence intervals.

lower odds of detection (OR, 0.78; 95% CI, 0.64–0.96; adjusted $p = 0.03$). Detection in the medium-injected-error condition did not differ significantly from the control condition (OR, 1.04; 95% CI, 0.81–1.34; adjusted $p = 0.76$). The omnibus test strongly rejects equality across conditions ($p < 0.01$), consistent with the predicted non-monotone relationship between injected-error prevalence and true-error detection. This pattern was robust across alternative model specifications (Table 5), sequential addition of covariates (Appendix Tables 12 and 13), and a supplemental analysis restricted to variables broadly important for assessing the causal relationship between a suspect drug and adverse event (Appendix Figure 12).

Having established support for Hypothesis 1, we next examine the mechanism underlying this result. We first test Hypotheses 1a and 1b by examining reviewers' review effort and trust in AI. We then test Hypothesis 1c by examining whether the improvement in true-error detection is accompanied by a corresponding increase in false-positive flagging. Finally, we turn to Hypothesis 2 and examine whether reviewer performance on injected errors predicts reviewer performance on naturally occurring AI errors.

6.2 Review Effort

Hypothesis 1a predicts that sufficiently frequent error injection increases review effort. We evaluate this prediction using three complementary measures of effort. Review time captures the amount of time devoted to each review, combined error detection captures the total number of error opportunities reviewers identify, and NASA-TLX measures reviewers' perceived workload.

6.2.1 Review Time

We first examine whether injected-error prevalence changes the amount of time reviewers devote to each review. Reviews lasting more than 30 minutes were excluded because they likely reflected

Table 5: **Association between injected-error condition and detection of true errors across model specifications.**

	Dependent variable: Detection of a true error		
	Document FE	Document + variable FE	Fully adjusted
Low injected errors	0.774^{**}	0.769^{**}	0.784^{**}
	$[0.624, 0.961]$	$[0.617, 0.958]$	$[0.638, 0.962]$
Medium injected errors	1.053	1.055	1.041
	$[0.812, 1.365]$	$[0.809, 1.377]$	$[0.809, 1.340]$
High injected errors	1.493^{***}	1.504^{***}	1.457^{***}
	$[1.206, 1.849]$	$[1.203, 1.881]$	$[1.181, 1.799]$
Study-period fixed effects	Yes	Yes	Yes
Cumulative review order	Yes	Yes	Yes
Reviewer fixed effects	Yes	Yes	Yes
Document fixed effects	Yes	Yes	Yes
Variable fixed effects	No	Yes	Yes
Baseline covariates × review order	No	No	Yes
Observations	20,669	20,669	20,669
McFadden pseudo-R^2	0.106	0.123	0.123
Omnibus condition p-value	<0.001	<0.001	<0.001

Notes: Odds ratios are reported with 95% confidence intervals. The control condition is the reference group. Standard errors are clustered at the reviewer level. Reviewer, document, and study-period fixed effects are included in all specifications. $^{*}p < 0.10$; $^{**}p < 0.05$; $^{***}p < 0.01$.

interruptions or inactivity rather than active review. Concretely, we utilized the following model:

$$\log(T_{idt}) = \beta_0 + \mathbf{C}'_{idt}\boldsymbol{\beta} + \theta O_{idt} + \gamma_t + \alpha_i + \delta_d + \varepsilon_{idt}, \tag{2}$$

where T_{idt} is review time in seconds for reviewer i reviewing document d during experimental period t. The vector $\mathbf{C}_{idt}$ contains indicators for the low-, medium-, and high-injected-error conditions; the omitted reference category is the control condition. O_{idt} denotes cumulative review order. The terms γ_t, α_i, and δ_d denote experimental-period, reviewer, and document fixed effects, respectively. Standard errors were clustered at the reviewer level. Figure 4 reports the raw mean review times and adjusted differences across injected-error conditions.

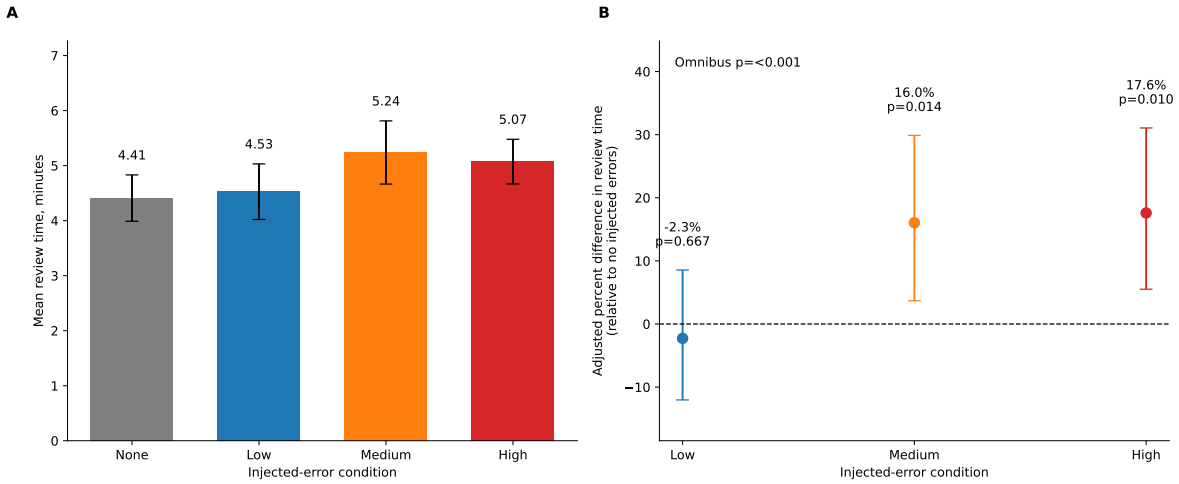

Figure 4: **Review time increases with injected-error frequency.** Panel A reports mean review times. Panel B reports adjusted percentage differences relative to the control condition. Error bars indicate 95% confidence intervals. p-values are corrected for multiple hypothesis testing.

Reviewers in the medium- and high-injected-error conditions spent substantially more time reviewing each article than reviewers in the control condition. In the raw data, mean review time increased from 4.4 minutes in the control condition to 5.2 and 5.1 minutes in the medium- and high-injected-error conditions, respectively. In contrast, review time in the low-injected-error condition was similar to that in the control condition (4.5 minutes).

The differences in review time were statistically significant in the omnibus test ($p < 0.001$). In the adjusted model, review time did not differ significantly between the low-injected-error and control conditions (review-time ratio, 0.98; 95% CI, 0.88–1.09; adjusted $p = 0.667$). Review time was significantly longer in the medium-injected-error condition than in the control condition (review-time ratio, 1.16; 95% CI, 1.04–1.30; adjusted $p = 0.015$), and significantly longer in the high-injected-error condition than in the control condition (review-time ratio, 1.18; 95% CI, 1.06–1.31; adjusted $p = 0.010$). These estimates correspond to 16.0% and 17.6% longer review times in the medium- and high-injected-error conditions, respectively. These results support Hypothesis 1a, indicating that reviewers in the medium- and high-injected-error conditions devoted more time to each review than reviewers in the control condition.

The increase in review time does not necessarily imply lower productivity, because reviewers in the high-injected-error condition also identified more true errors. In Appendix J.3, we show that

true-error detection efficiency was marginally higher in the high-injected-error condition than in the control condition, although this difference is not statistically significant.

6.2.2 Combined Error Detection

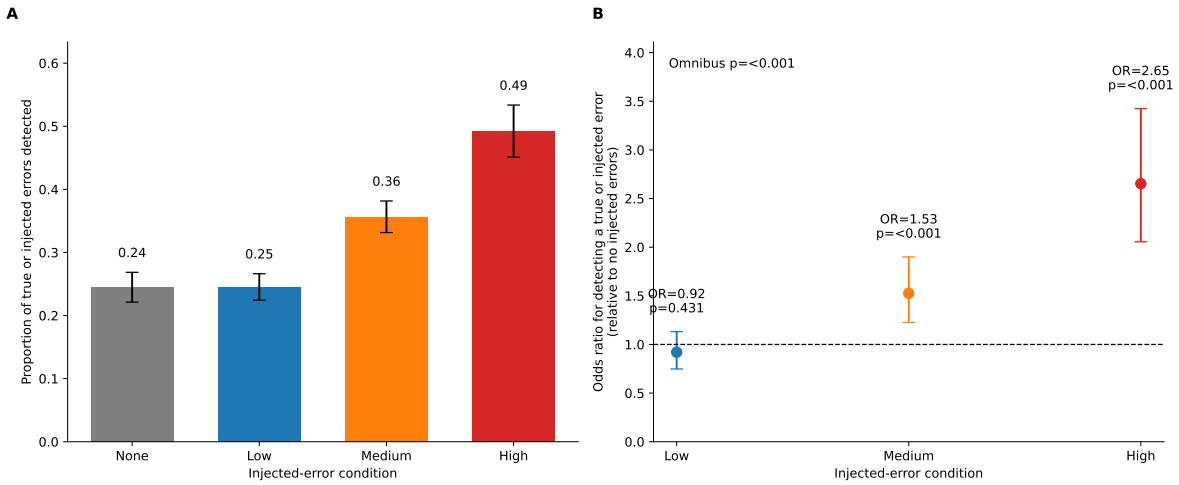

Figure 5: Combined detection of true and injected errors by injected-error condition. Panel A shows the raw proportion of true errors and injected errors detected. Panel B shows fully adjusted odds ratios for detecting either a true error or an injected error relative to the no-injection condition. Error bars indicate 95% confidence intervals.

We now examine a complementary measure of reviewer effort by looking at reviewers' detection of all available error opportunities, including injected and naturally occurring, in Figure 5. In this combined-error analysis, overall error detection differed significantly across conditions (omnibus $p < 0.01$). Raw combined detection rates were 24.5% in the no-injection condition, 24.5% in the low-injection condition, 35.7% in the medium-injection condition, and 49.2% in the high-injection condition. In the fully adjusted model, low-frequency error injection was not statistically different from no injection for overall error detection (OR, 0.92; 95% CI, 0.75–1.13; adjusted $p = 0.43$), even though it significantly reduced detection of naturally occurring true errors. By contrast, medium-frequency error injection increased overall error detection (OR, 1.53; 95% CI, 1.23–1.90; adjusted $p < 0.01$), and high-frequency error injection produced the largest increase (OR, 2.65; 95% CI, 2.06–3.42; adjusted $p < 0.01$). These results complement the review-time findings. Low-frequency injection did not significantly change overall error detection despite reducing detection of naturally occurring AI errors, whereas medium- and high-frequency injection increased reviewers' overall detection of errors.

6.2.3 Perceived Workload

The preceding analyses show that reviewers in the medium- and high-injected-error conditions spent more time reviewing documents and identified more error opportunities overall. We next examine whether these differences were accompanied by differences in perceived workload. Figure 6 examines whether these differences in review effort were accompanied by differences in perceived workload.

Perceived workload was measured using the NASA Task Load Index (NASA-TLX) and defined as the mean of six seven-point items assessing mental demand, physical demand, time pressure, effort, stress, and perceived task success. The task-success item was reverse coded so that higher scores consistently indicated greater workload. The full survey instruments are provided in Appendix I. We estimated:

$$L_{it} = \beta_0 + \mathbf{C}'_{it}\boldsymbol{\beta} + \theta O_{it} + \gamma_t + \alpha_i + \varepsilon_{it}, \tag{3}$$

where $\mathbf{C}_{it}$ contains injected-error-condition indicators, O_{it} denotes cumulative review order, and γ_t and α_i denote week and reviewer fixed effects, respectively. Standard errors were clustered at the reviewer level. Figure 6 reports adjusted differences relative to the control condition. Perceived

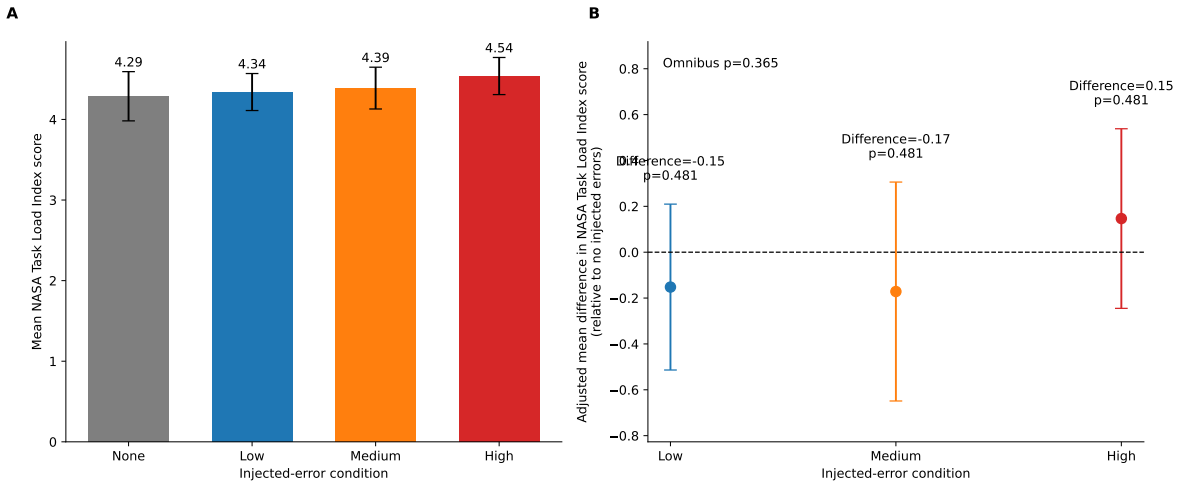

Figure 6: **Perceived workload remains stable across injected-error frequencies.** Panel A reports mean NASA-TLX scores. Panel B reports adjusted mean differences relative to the control condition. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

workload was similar across all four injected-error conditions. In the raw data, mean NASA-TLX scores increased only slightly from 4.29 in the control condition to 4.34, 4.39, and 4.54 in the low-, medium-, and high-injected-error conditions, respectively. The omnibus test did not reject equality across conditions ($p = 0.365$).

The adjusted results lead to the same conclusion. Mean NASA-TLX scores did not differ significantly between the low-injected-error and control conditions (adjusted mean difference, −0.15; 95% CI, −0.51 to 0.21; adjusted $p = 0.481$). Task load also did not differ significantly between the medium-injected-error and control conditions (adjusted mean difference, −0.17; 95% CI, −0.65 to 0.31; adjusted $p = 0.481$) or between the high-injected-error and control conditions (adjusted mean difference, 0.15; 95% CI, −0.24 to 0.54; adjusted $p = 0.481$).

Overall, these findings support Hypothesis 1a based on the two objective measures of review effort: review time and overall error detection. Although raw perceived workload also increased, the effect was not statistically significant, suggesting that any increase in subjective workload was modest relative to the changes observed in the objective measures.

6.3 Trust in AI

Hypothesis 1b predicts that sufficiently frequent error injection reduces reviewers' trust in AI-generated summaries, which in turn leads reviewers to spend more time verifying AI-generated outputs. To evaluate this prediction, we measured reviewer trust throughout the experiment. Trust was defined as the mean of two five-point Likert-scale items assessing general trust in AI-system outputs and trust in AI-generated summaries of adverse drug event reports. Change in trust was defined relative to each reviewer's trust score before the start of the experiment. We estimated:

$$\Delta T_{it} = \beta_0 + \mathbf{C}'_{it}\boldsymbol{\beta} + \theta O_{it} + \gamma_t + \alpha_i + \varepsilon_{it}, \tag{4}$$

where $\mathbf{C}_{it}$ contains injected-error-condition indicators, O_{it} denotes cumulative review order, and γ_t and α_i denote week and reviewer fixed effects, respectively. Standard errors were clustered at the reviewer level. Figure 7 reports raw mean changes and adjusted differences relative to the control condition.

Trust remained approximately unchanged in the control condition but declined as injected-error frequency increased. In the raw data, mean trust changed by approximately 0.02 relative to baseline in the control condition, compared with −0.18, −0.09, and −0.31 in the low-, medium-, and high-injected-error conditions, respectively. The largest decline occurred in the high-injected-error condition.

The adjusted results show a similar pattern. Change in trust did not differ significantly between the low-injected-error and control conditions (adjusted mean difference, −0.17; 95% CI, −0.39 to 0.06; adjusted $p = 0.212$) or between the medium-injected-error and control conditions (adjusted mean difference, −0.08; 95% CI, −0.29 to 0.13; adjusted $p = 0.455$). However, reviewers in the high-injected-error condition experienced a significantly greater decline in trust than reviewers in the control condition (adjusted mean difference, −0.27; 95% CI, −0.49 to −0.05; adjusted $p = 0.047$).

These findings suggest that reviewers exposed to high levels of injected errors become less trusting of the AI-generated output. Consistent with this interpretation, reviewers in the high-injected-error condition spend more time reviewing summaries and identify more true errors than reviewers in the control condition. The decline in trust therefore appears to co-occur with increased vigilance during review.

6.4 False Positive Detection

Hypothesis 1c predicts that improvements in true-error detection do not arise from reviewers becoming indiscriminately more likely to flag potential errors. To evaluate this prediction, we examined false-positive flagging across injected-error conditions. A false positive was defined as a variable flagged by a reviewer despite containing neither a true error nor an injected error. We estimated the same fully adjusted logistic regression specification shown in Equation 1, replacing the outcome with an indicator equal to 1 if reviewer i incorrectly flagged variable v of document d during experimental period t. Figure 8 reports the resulting odds ratios across injected-error conditions.

False-positive flagging was similar across all four injected-error conditions, and the omnibus test did not reject equality across conditions ($p = 0.584$). In the raw data, false-positive rates remained largely unchanged as the frequency of injected errors increased.

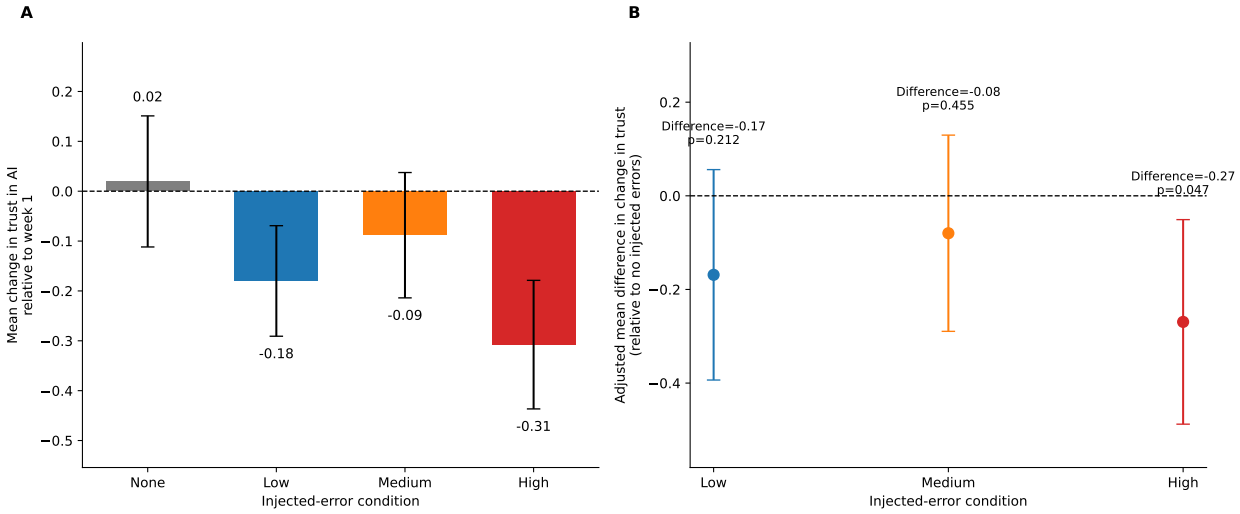

Figure 7: **Trust in AI declines at high levels of injected-error frequency.** Panel A reports raw changes in trust relative to baseline. Panel B reports adjusted mean differences relative to the control condition. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

The regression results lead to the same conclusion. In the fully adjusted model, the odds of falsely flagging a correct cell were not significantly different in the low-injected-error condition compared with the control condition (OR, 1.10; 95% CI, 0.91–1.33; adjusted $p = 0.320$). False-positive flagging also did not differ significantly between the medium-injected-error and control conditions (OR, 1.15; 95% CI, 0.92–1.43; adjusted $p = 0.320$) or between the high-injected-error and control conditions (OR, 1.14; 95% CI, 0.92–1.41; adjusted $p = 0.320$).

These findings support Hypothesis 1c. The differences in true-error detection do not appear to be driven by reviewers becoming more willing to flag potential errors. Reviewers exposed to higher levels of injected errors identified more true errors without becoming more likely to incorrectly flag correct information.

6.5 Auditing Reviewer Performance

Hypothesis 2 predicts that reviewer-level detection of injected errors is positively associated with reviewer-level detection of naturally occurring AI errors. We evaluate this prediction by examining whether reviewers who identify a larger proportion of injected errors also identify a larger proportion of naturally occurring AI errors. If so, performance on injected errors may provide a practical mechanism for monitoring reviewer performance without requiring an independent audit of every review.

Figure 9 reports the relationship between reviewer-level detection of injected errors and reviewer-level detection of true errors. Reviewers who were more likely to identify injected errors were also more likely to identify true AI-generated errors. Across 17 reviewers, the Pearson correlation between true-error detection rates and injected-error detection rates was $r = 0.55$ ($p = 0.02$). The association was similar using the Spearman rank correlation ($\rho = 0.64$; $p = 0.01$). Reviewers with the highest injected-error detection rates tended to have substantially higher true-error detection

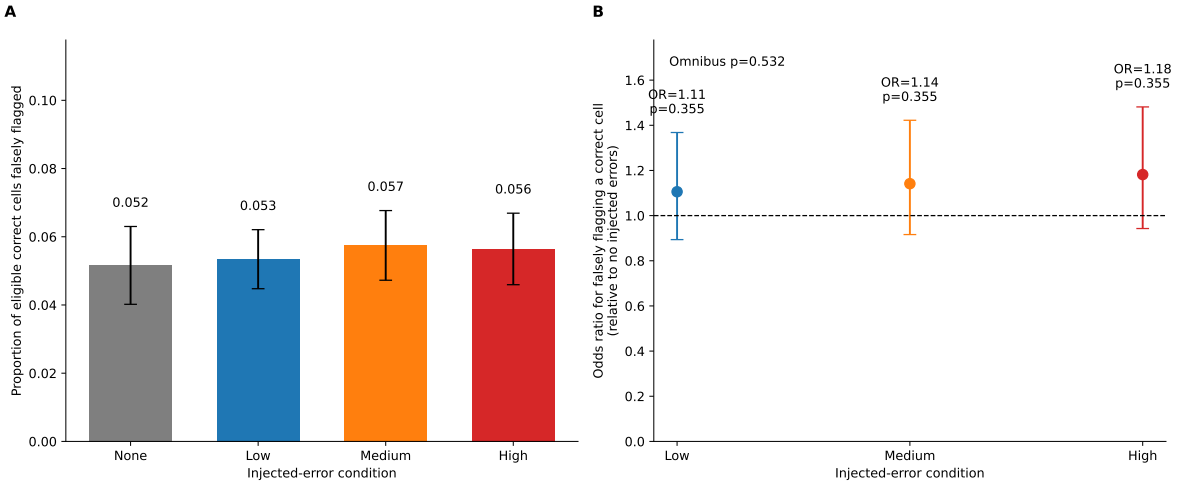

Figure 8: **False-positive flagging does not vary with injected-error frequency.** Panel A reports raw false-positive rates. Panel B reports adjusted odds ratios relative to the control condition. Error bars indicate 95% confidence intervals. p-values are corrected for multiple hypothesis testing.

rates than reviewers with the lowest injected-error detection rates.

These findings suggest that injected-error detection provides information about reviewers' underlying ability to identify true errors. Because injected errors are observable to the organization by design, they may provide a practical mechanism for monitoring reviewer performance without requiring independent verification of naturally occurring AI errors.

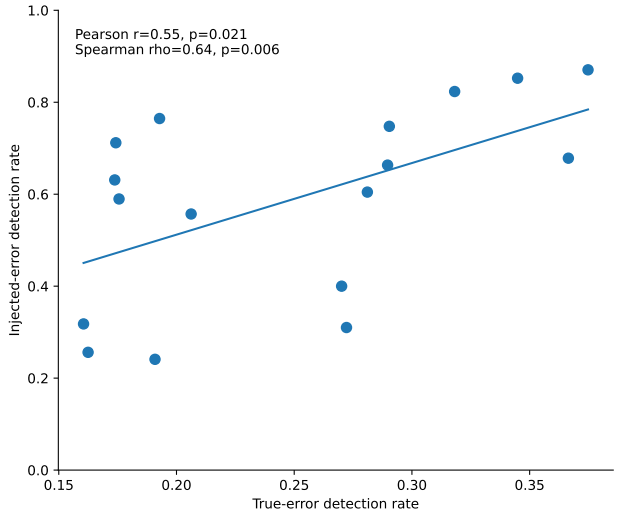

Figure 9: **Reviewers who identify more injected errors also identify more true errors.** Each point represents one reviewer. The horizontal axis reports true-error detection rates and the vertical axis reports injected-error detection rates.

6.6 Changes in Human Behavior over Time

The analyses above estimate average treatment effects across experimental periods. We next examine whether reviewer behavior changes within a period after reviewers have spent additional time under the same injection condition. Because most within-period changes are not statistically significant, we report the detailed outcome-specific estimates in Appendix J.5 and summarize the main patterns here.

For each reviewer and experimental period, we order completed reviews sequentially and divide them into quartiles. We compare reviews completed in the fourth quartile with reviews completed in the first quartile, using only observations from these two quartiles. The adjusted models take the form

$$g(Y_{idvt}) = \beta_0 + \mathbf{C}'_{idvt}\boldsymbol{\beta} + \rho Q_{it} + \mathbf{C}'_{idvt}\boldsymbol{\kappa} Q_{it} + \gamma_t + \alpha_i + \delta_d + \varepsilon_{idvt}, \tag{5}$$

where Q_{it} is an indicator equal to 1 for reviews completed in the fourth quartile of reviewer i's review order within experimental period t and equal to 0 for reviews completed in the first quartile. The vector $\mathbf{C}_{idvt}$ contains indicators for the low-, medium-, and high-injected-error conditions, with the control condition as the omitted reference category. The terms γ_t, α_i, and δ_d denote experimental-period, reviewer, and document fixed effects, respectively. For binary outcomes, $g(\cdot)$ is the logit link. For review time, the outcome is the natural logarithm of review time, and the model is estimated by ordinary least squares. Standard errors are clustered at the reviewer level.

Figure 16 reports within-period changes in review time. In the control condition, review time was 24.3% lower in the fourth quartile than in the first quartile (95% CI, −34.3% to −12.7%; adjusted $p < 0.01$). Similar declines were observed in the low-injected-error condition (−24.2%; 95% CI, −35.3% to −11.0%; adjusted $p < 0.01$) and the high-injected-error condition (−23.9%; 95% CI, −32.3% to −14.6%; adjusted $p < 0.01$). Review time also declined in the medium-injected-error condition, although the estimate was smaller and not statistically significant (−10.9%; 95% CI, −23.9% to 4.3%; adjusted $p = 0.15$). The interaction between injected-error condition and within-period review order was not statistically significant (interaction omnibus $p = 0.08$). These results suggest that reviewers completed reviews more quickly later in a review period, but that the magnitude of this reduction did not differ significantly across injected-error conditions.

Other outcomes exhibited little evidence of systematic within-period change. True-error detection, trust in AI, and the association between injected-error detection and true-error detection did not change significantly over the course of a review period. False-positive flagging increased within the high-injected-error condition, but no comparable changes were observed in the remaining conditions. These exploratory analyses suggest that the primary treatment effects are generally stable over the three-week intervention period. Reviewers became faster over time, but we do not observe corresponding changes in accuracy or trust.

6.7 Discussion

Error injection interventions require calibration to work well. Our findings suggest that controlled error injection can improve human oversight in AI-assisted review workflows, but only when reviewers encounter injected errors frequently enough. Low levels of error injection reduced reviewers' ability to identify naturally occurring AI errors relative to the control condition, whereas high levels substantially improved true-error detection. The contrast between these conditions

suggests that occasional exposure to injected errors adds burden without changing reviewer behavior, whereas frequent exposure induces more careful review. Appendix A formalizes this intuition utilizing a simple stylized model. The model shows how low levels of error injection can reduce performance by increasing review burden without increasing vigilance, whereas high levels can improve performance by reducing trust in the AI output and increasing the return to vigilance.

One interpretation is that error injection works by calibrating reviewers' perceived error prevalence. In our study, naturally occurring true errors appeared in roughly 10% of article-variable cells, whereas injected errors appeared in approximately 1%, 5%, and 15% of cells in the low-, medium-, and high-injection conditions, respectively. The high-injection condition was therefore the only condition in which the injected-error rate exceeded the naturally occurring true-error rate. Under this interpretation, the low- and medium-injection conditions may have provided too few error signals to shift reviewers' expectations about how often AI-generated summaries were likely to be wrong. By contrast, the high-injection condition may have made errors sufficiently frequent and salient to increase reviewers' perceived error prevalence, leading them to scrutinize AI-generated summaries more closely.

This finding has broader implications for the design of interventions intended to sustain vigilance in AI-assisted workflows. A natural intuition is that any mechanism that reminds reviewers that AI systems can make mistakes should improve oversight. Our results suggest that this intuition is incomplete. Sparse exposure to injected errors appears insufficient to alter reviewer behavior meaningfully and may even reduce performance relative to no intervention at all. In contrast, more frequent exposure to injected errors appears to fundamentally change how reviewers interact with AI-generated outputs. The implication is that vigilance interventions may require calibration. The question is not simply whether to intervene, but how intensively to intervene and whether the injected-error rate is high enough to shape reviewers' expectations about the underlying error environment.

The evidence is consistent with increased vigilance through calibrated distrust. The evidence supporting Hypotheses 1a–1c is consistent with a shift in reviewer vigilance. Reviewers exposed to high levels of injected errors spent more time reviewing summaries, experienced a decline in trust in the AI system, and identified substantially more true errors. At the same time, false-positive rates remained stable and perceived workload did not increase significantly. Viewed together, these results suggest that reviewers in the high-injection condition were not merely flagging more fields indiscriminately. Instead, they appear to have adopted a more careful review strategy characterized by greater scrutiny of AI-generated outputs. The decline in trust observed under high levels of error injection is consistent with this interpretation. In high-stakes review environments, the goal may not be to maximize trust in AI systems, but rather to maintain appropriately calibrated skepticism.

The contrast between the low- and high-injection conditions further suggests that reviewer vigilance is shaped by the perceived prevalence of AI errors. Reviewers in the low-injection condition encountered enough injected errors to be exposed to the intervention but not enough to exhibit the behavioral changes observed in the high-injection condition. One interpretation is that infrequent injected errors reinforce the perception that errors are rare and easily identified, creating a false sense of security. More frequent injected errors may instead keep the possibility of error salient throughout the review process. Although our experiment was not designed to isolate the precise psychological

mechanism, the combined patterns in trust, review effort, and true-error detection are consistent with this explanation. Moreover, we found little evidence that these outcomes changed systematically within an experimental period, suggesting that the differences across injection conditions are unlikely to be explained by reviewers gradually learning to detect errors over time.

Error injection can serve both as an intervention and as a measurement system. Our findings also highlight a second role for controlled error injection. In many AI-assisted review workflows, organizations cannot directly observe whether reviewers successfully identify naturally occurring AI errors. Throughput and turnaround time are observable, but oversight quality is often not. Because injected errors have known ground truth, they create review opportunities that can be evaluated directly. Consistent with Hypothesis 2, reviewers who identified a greater fraction of injected errors also identified a greater fraction of naturally occurring AI errors. This relationship suggests that injected-error detection may provide a practical signal of reviewer vigilance and creates the possibility of monitoring review quality without requiring a second independent audit of every case. In this sense, controlled error injection functions as both an intervention for changing reviewer behavior and a measurement system for assessing oversight quality.

More broadly, our results suggest that human oversight in AI-assisted workflows should be viewed as a design problem rather than a fixed property of the workflow itself. Organizations often assume that placing a human reviewer after an AI system is sufficient to ensure oversight. Our findings suggest otherwise. Reviewer behavior depends on the environment in which review occurs, including how frequently reviewers encounter errors, how much effort is required to verify outputs, and how they calibrate trust in the underlying AI system. Interventions that alter these conditions can substantially affect oversight quality.

Although our experiment was conducted in pharmacovigilance review, the underlying challenge extends to a broad class of AI-assisted workflows in which humans verify model-generated drafts before downstream use. Similar structures appear in medical documentation, insurance claims review, content moderation, customer support, and other review-intensive settings. Across these environments, organizations increasingly rely on human review as a safeguard against AI error while simultaneously facing difficulty measuring whether that review remains effective. Controlled error injection offers one approach to addressing both the vigilance problem and the measurement problem.

7 Limitations

Several limitations should be considered when interpreting our findings. First, the experiment was conducted with PharmD students performing a simulated pharmacovigilance review task. Although the platform was designed to resemble an operational AI-assisted workflow and participants possessed relevant clinical training, reviewer behavior may differ among experienced pharmacovigilance professionals operating in production environments.

Second, injected errors were designed to resemble realistic AI-generated inaccuracies, but synthetic errors may differ from naturally occurring model failures in their salience and complexity. This difference may help explain why injected errors were detected at higher rates than naturally occurring true errors. Because injected errors were introduced programmatically and used as known ground-truth opportunities for feedback, they were generally discrete changes to a specific summary

field. By contrast, naturally occurring AI errors were sometimes more embedded in the case context and may have required reviewers to integrate information across multiple parts of the report. Thus, injected-error detection should not be interpreted as equivalent to true-error detection. Instead, injected errors may function as a calibrated vigilance cue while remaining an imperfect proxy for naturally occurring model failures. Future work should examine whether the effects of error injection vary across error categories, levels of subtlety, and clinical severity.

Third, our experiment evaluates reviewer behavior over a twelve-week period. Although we find little evidence of within-period fatigue and observe stable treatment effects throughout the study, longer-term responses remain uncertain. Reviewers may eventually adapt to the presence of injected errors, organizations may modify feedback procedures over time, and the optimal injection frequency may depend on the duration of deployment. Evaluating these longer-run dynamics represents an important direction for future research.

Fourth, although high-frequency error injection improved true-error detection, it did not eliminate missed errors. Reviewers in the high-injection condition still identified only about one-third of visible true errors, meaning that approximately two-thirds remained undetected. Controlled error injection should therefore be viewed as one component of an oversight strategy rather than a complete solution. In high-stakes workflows, organizations may need to combine error injection with other safeguards, such as targeted auditing, improved interface design, reviewer training, escalation rules for high-risk cases, or automated checks for specific error types. The acceptable residual miss rate will likely depend on the clinical or operational consequences of different errors, making calibration of both injection frequency and complementary safeguards an important implementation decision.

Finally, our auditing analysis should be interpreted cautiously. Although reviewers who identified more injected errors also identified more true AI-generated errors, the correlation is estimated using a relatively small number of reviewers. Additional field studies are needed to determine whether injected-error detection can reliably serve as a proxy for oversight quality in operational settings.

8 Conclusion

Organizations increasingly rely on human review as a safeguard against errors in AI-generated outputs. Yet the effectiveness of this safeguard depends on reviewers remaining attentive even when AI systems become highly accurate and errors become increasingly rare. In many settings, organizations also lack scalable mechanisms for determining whether reviewers are meaningfully verifying AI-generated content.

This paper studies controlled error injection as a mechanism for addressing both challenges. In a preregistered longitudinal experiment in AI-assisted pharmacovigilance review, we show that the effect of error injection depends critically on the frequency with which errors are injected. Low levels of error injection reduced reviewers' ability to identify naturally occurring AI errors, whereas high levels of error injection substantially improved true-error detection. The broader pattern of results suggests that higher levels of error injection increase reviewer vigilance: reviewers spent more time reviewing summaries, exhibited lower trust in the AI-generated output, and identified more true errors without increasing false-positive rates. At the same time, reviewers who detected more injected errors also detected more naturally occurring errors, suggesting that error injection may provide a practical mechanism for monitoring oversight quality.

More broadly, our findings suggest that effective human oversight is not guaranteed simply by placing a human reviewer after an AI system. Reviewer behavior depends on how the review environment is designed, including how frequently reviewers encounter errors and how they calibrate trust in the underlying model. As AI-generated drafts become increasingly common across healthcare, finance, insurance, customer support, and other review-intensive domains, organizations will face growing pressure to ensure that human oversight remains effective. Controlled error injection represents one possible approach. Our results suggest that it can improve oversight, but only when organizations recognize that the frequency of injected errors is itself a consequential design decision.

References

Nikhil Agarwal, Alex Moehring, Pranav Rajpurkar, and Tobias Salz. Combining human expertise with artificial intelligence: Experimental evidence from radiology. Working Paper 31422, National Bureau of Economic Research, 2023.

Lisanne Bainbridge. Ironies of automation. *Automatica*, 19(6):775–779, 1983. doi: 10.1016/0005-1098(83)90046-8.

Maya Balakrishnan, Kris Johnson Ferreira, and Jordan Tong. Human-algorithm collaboration with private information: Naïve advice-weighting behavior and mitigation. *Management Science*, 72 (1):265–284, 2026.

Hamsa Bastani and Gerard P. Cachon. The human-ai contracting paradox. *Available at SSRN*, 2025. SSRN working paper.

Leonard Boussioux, Jacqueline N. Lane, Miaomiao Zhang, Vladislav Jacimovic, and Karim R. Lakhani. The crowdless future? generative ai and creative problem-solving. *Organization Science*, 35(5):1589–1607, 2024. doi: 10.1287/orsc.2023.18430.

Erik Brynjolfsson, Danielle Li, and Lindsey R. Raymond. Generative ai at work. *The Quarterly Journal of Economics*, 140(2):889–942, 2025. doi: 10.1093/qje/qjae044.

D. Buser, Adrian Schwaninger, V. Rehor, and Yanik Sterchi. Reliability and validity of threat image projection data as a measure of performance in x-ray baggage screening. *Transportation Research Part A: Policy and Practice*, 200:104640, 2025. doi: 10.1016/j.tra.2025.104640.

Fabrizio Dell'Acqua, Edward McFowland, Ethan R. Mollick, Hila Lifshitz-Assaf, Katherine C. Kellogg, Saran Rajendran, Lisa Krayer, François Candelon, and Karim R. Lakhani. Navigating the jagged technological frontier: Field experimental evidence of the effects of ai on knowledge worker productivity and quality. Working Paper 24-013, Harvard Business School, 2023.

Kate Dowdy, Anna Hoffman, Tyree Giles, David Kugele, Jacqueline Guill, Isaac Chang, and Gregory Jackson. Summarizing faers narratives with generative ai: Methods, resource requirements, and quality assessment. FDA Scientific Computing and Digital Transformation Symposium, 2024. Booz Allen Hamilton and Center for Drug Evaluation and Research, U.S. Food and Drug Administration. Available at `https://www.fda.gov/media/182798/download`.

Nicholas S. Downing, Nilay D. Shah, Jenerius A. Aminawung, Alison M. Pease, Jean-David Zeitoun, Harlan M. Krumholz, and Joseph S. Ross. Postmarket safety events among novel therapeutics approved by the us food and drug administration between 2001 and 2010. *JAMA*, 317(18): 1854–1863, 2017. doi: 10.1001/jama.2017.5150.

Kate Goddard, Abdul Roudsari, and Jeremy C. Wyatt. Automation bias: A systematic review of frequency, effect mediators, and mitigators. *Journal of the American Medical Informatics Association*, 19(1):121–127, 2012. doi: 10.1136/amiajnl-2011-000089.

Julien Grand-Clément and Jean Pauphilet. The best decisions are not the best advice: Making adherence-aware recommendations. *Management Science*, 72(1):667–692, 2026.

Joe B. Hakim, Jeffery L. Painter, Darmendra Ramcharran, Vijay Kara, Greg Powell, Paulina Sobczak, Chiho Sato, Andrew Bate, and Andrew Beam. The need for guardrails with large language models in pharmacovigilance and other medical safety critical settings. *Scientific Reports*, 15:27886, 2025. doi: 10.1038/s41598-025-09138-0.

Sandra G. Hart and Lowell E. Staveland. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In Peter A. Hancock and Najmedin Meshkati, editors, *Human Mental Workload*, volume 52 of *Advances in Psychology*, pages 139–183. North-Holland, Amsterdam, 1988. doi: 10.1016/S0166-4115(08)62386-9.

Franziska Hofer and Adrian Schwaninger. Using threat image projection data for assessing individual screener performance. In *Safety and Security Engineering II*, pages 417–426. WIT Press, 2007. doi: 10.2495/SAFE070401.

Vijay Kara, Greg Powell, Olivia Mahaux, Aparna Jayachandra, Naashika Nyako, Christopher Golds, and Andrew Bate. Finding needles in the haystack: Clinical utility score for prioritisation (cusp), an automated approach for identifying spontaneous reports with the highest clinical utility. *Drug Safety*, 46(9):847–855, 2023. doi: 10.1007/s40264-023-01327-y.

Saravanan Kesavan and Tarun Kushwaha. Field experiment on the profit implications of merchants' discretionary power to override data-driven decision-making tools. *Management Science*, 66(11): 5182–5198, 2020.

Caleb Kwon, Ananth Raman, and Jorge Tamayo. Human-computer interactions in demand forecasting and labor scheduling decisions. Working paper, Harvard Business School, 2022.

Markus Langer, Kevin Baum, and Nadine Schlicker. Effective human oversight of ai-based systems: A signal detection perspective on the detection of inaccurate and unfair outputs. *Minds and Machines*, 35(1):1, 2025. doi: 10.1007/s11023-024-09701-0.

Sarah Lebovitz, Hila Lifshitz-Assaf, and Natalia Levina. To engage or not to engage with ai for critical judgments: How professionals deal with opacity when using ai for medical diagnosis. *Organization Science*, 33(1):126–148, 2022. doi: 10.1287/orsc.2021.1549.

Thodoris Lykouris and Wentao Weng. Learning to defer in congested systems: The ai-human interplay. *arXiv preprint arXiv:2402.12237*, 2025. doi: 10.48550/arXiv.2402.12237.

N. H. Mackworth. The breakdown of vigilance during prolonged visual search. *Quarterly Journal of Experimental Psychology*, 1(1):6–21, 1948. doi: 10.1080/17470214808416738.

Raja Parasuraman and Dietrich H. Manzey. Complacency and bias in human use of automation: An attentional integration. *Human Factors*, 52(3):381–410, 2010. doi: 10.1177/0018720810376055.

Robin Riz à Porta, Yanik Sterchi, and Adrian Schwaninger. How realistic is threat image projection for x-ray baggage screening? *Sensors*, 22(6):2220, 2022. doi: 10.3390/s22062220.

Jeremy Schwark, Joshua Sandry, and Igor Dolgov. False feedback increases detection of low-prevalence targets in visual search. *Attention, Perception, & Psychophysics*, 74(8):1583–1589, 2012. doi: 10.3758/s13414-012-0354-4.

Jiankun Sun, Dennis J. Zhang, Haoyuan Hu, and Jan A. Van Mieghem. Predicting human discretion to adjust algorithmic prescription: A large-scale field experiment in warehouse operations. *Management Science*, 68(2):846–865, 2022.

J. Eric T. Taylor, Matthew D. Hilchey, Blaire J. Weidler, and Jay Pratt. Eliminating the low-prevalence effect in visual search with a remarkably simple strategy. *Psychological Science*, 33(5): 717–724, 2022. doi: 10.1177/09567976211048485.

U.S. Food and Drug Administration. FDA Adverse Event Reporting System (FAERS) Database. `https://www.fda.gov/drugs/drug-approvals-and-databases/fda-adverse-event-reporting-system-faers-database`, 2017. Content current as of June 9, 2017; accessed May 26, 2026.

U.S. Food and Drug Administration. Postmarketing Adverse Event Reporting: Required Information. `https://www.fda.gov/drugs/surveillance/postmarketing-adverse-event-reporting-required-information`, 2018. Accessed May 26, 2026.

U.S. Food and Drug Administration. Postmarketing Surveillance Programs. `https://www.fda.gov/drugs/surveillance/postmarketing-surveillance-programs`, 2025. Accessed May 26, 2026.

Michelle Vaccaro, Abdullah Almaatouq, and Thomas W. Malone. When combinations of humans and ai are useful: A systematic review and meta-analysis. *Nature Human Behaviour*, 8:2293–2303, 2024. doi: 10.1038/s41562-024-02024-1.

Jen Van Tiem, Elizabeth Cramer, Christopher Iverson, Korey Kennelty, Noah Andrys, Julie Lee, Lindsey Knake, Jason Misurac, James Blum, and Heather Schacht Reisinger. Listening to the note: Clinician perspectives on ambient artificial intelligence scribes in medical documentation. *Journal of the American Medical Informatics Association*, 33(2):255–262, 2026. doi: 10.1093/jamia/ocaf214.

Joel S. Warm, Raja Parasuraman, and Gerald Matthews. Vigilance requires hard mental work and is stressful. *Human Factors*, 50(3):433–441, 2008. doi: 10.1518/001872008X312152.

E. J. Williams. Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Scientific Research*, 2(2):149–168, 1949. doi: 10.1071/CH9490149.

Jeremy M. Wolfe, Todd S. Horowitz, and Naomi M. Kenner. Rare items often missed in visual searches. *Nature*, 435(7041):439–440, 2005. doi: 10.1038/435439a.

Jeremy M. Wolfe, Todd S. Horowitz, Michael J. Van Wert, Naomi M. Kenner, Susan S. Place, and Nora Kibbi. Low target prevalence is a stubborn source of errors in visual search tasks. *Journal of Experimental Psychology: General*, 136(4):623–638, 2007. doi: 10.1037/0096-3445.136.4.623.

Jeremy M. Wolfe, David N. Brunelli, Joshua Rubinstein, and Todd S. Horowitz. Prevalence effects in newly trained airport checkpoint screeners: Trained observers miss rare targets, too. *Journal of Vision*, 13(3):33, 2013. doi: 10.1167/13.3.33.

Qichuan Yin, Ziwei Su, and Shuangning Li. Overcoming the incentive collapse paradox. *arXiv preprint arXiv:2603.27049*, 2026.

Online Supplement: Error Injection for AI Oversight

A Stylized Model for Error Injection

A.1 Individual Reviewer Vigilance

This appendix presents a stylized model for controlled error injection that formalizes the theoretical predictions in Section 4. Reviewers receive a fixed payment together with an accuracy based bonus. The payment rule is identical across injection conditions. Error injection therefore does not change the financial incentive directly. It changes the review environment in which reviewers evaluate AI-generated summaries.

Consider reviewer i in an AI-assisted review workflow. Let W denote the fixed payment and let $B > 0$ denote the monetary weight placed on review accuracy. If reviewer i achieves percentage accuracy A_i, total compensation is $W + BA_i$. The reviewer chooses a hidden vigilance level $e_i \in \{0, 1\}$, where $e_i = 0$ denotes routine review and $e_i = 1$ denotes vigilant review. Vigilant review requires additional attention and incurs cost $c_i > 0$.

Each field is either correct, contains a naturally occurring AI error, or contains an injected error. Let p denote the prevalence of naturally occurring AI errors, and let q denote the prevalence of intentionally injected errors, with $p + q \leq 1$. The case $q = 0$ corresponds to no error injection. Reviewers need not know the value of q. The parameter q captures the error environment reviewers experience through repeated review.

If a field contains a naturally occurring AI error, the probability that reviewer i detects the error is

$$s_N(e_i) = s_N^0 + \Delta_N e_i,$$

where $\Delta_N > 0$. If a field contains an injected error, the probability that reviewer i detects the error is

$$s_I(e_i) = s_I^0 + \Delta_I e_i,$$

where $\Delta_I > 0$. If a field is correct, the probability that the reviewer correctly leaves the field unflagged is

$$z(e_i) = z_0 + \Delta_z e_i.$$

The parameters Δ_N and Δ_I allow vigilant review to have different returns for naturally occurring and injected errors. Injected errors also impose an attentional cost on detection of naturally occurring AI errors. Let $\ell > 0$ measure this cost. We assume parameters are such that all probabilities lie in $[0, 1]$ over the relevant range of q.

The reviewer's expected accuracy under injected error prevalence q is

$$A(e_i, q) = p\{s_N(e_i) - \ell q\} + q s_I(e_i) + (1 - p - q) z(e_i).$$

Reviewer i chooses vigilance to maximize

$$U_i(e_i; q) = W + BA(e_i, q) - c_i e_i.$$

The incremental utility from vigilant review is

$$\begin{aligned} U_i(1;q) - U_i(0;q) &= B\left[p\{s_N(1) - s_N(0)\} + q\{s_I(1) - s_I(0)\} + (1-p-q)\{z(1) - z(0)\}\right] - c_i \\ &= B\left[p\Delta_N + q\Delta_I + (1-p-q)\Delta_z\right] - c_i \\ &= B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right] - c_i. \end{aligned}$$

Reviewer i chooses vigilant review if and only if

$$B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right] \geq c_i.$$

Proposition 1. *Suppose $\Delta_I > \Delta_z$. Then the return to vigilant review is increasing in injected error prevalence q. For each reviewer i, there exists a threshold*

$$q_i^* = \frac{c_i/B - \Delta_z - p(\Delta_N - \Delta_z)}{\Delta_I - \Delta_z}$$

such that the reviewer chooses vigilant review if and only if $q \geq q_i^*$, whenever $q_i^* \in (0, 1-p)$.

Proof. Define the net return to vigilance as

$$G_i(q) = B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right] - c_i.$$

Since

$$G_i'(q) = B(\Delta_I - \Delta_z) > 0,$$

the net return to vigilance is strictly increasing in injected error prevalence. Solving $G_i(q_i^*) = 0$ gives the stated threshold. Since $G_i(q)$ is increasing, reviewer i chooses vigilant review if and only if $q \geq q_i^*$. □

Proposition 1 characterizes how injected error prevalence changes the return to vigilant review. As injected errors become more prevalent, the same accuracy based incentive makes vigilant review optimal for a larger set of reviewers.

Let $D_i(q)$ denote reviewer i's probability of detecting a naturally occurring AI error under injected error prevalence q. The detection function implied by the model is

$$D_i(q) = s_N(e_i^*(q)) - \ell q = s_N^0 + \Delta_N e_i^*(q) - \ell q,$$

where $e_i^*(q)$ is the optimal vigilance choice characterized above.

Proposition 2. *Suppose reviewer i has threshold q_i^* and suppose $0 < q_L < q_i^* \leq q_H \leq 1 - p$. Then low prevalence injection reduces naturally occurring AI error detection relative to no-injection:*

$$D_i(q_L) < D_i(0).$$

High prevalence injection improves naturally occurring AI error detection relative to no injection if and only if

$$\Delta_N > \ell q_H.$$

Proof. Because $0 < q_L < q_i^*$, both $q = 0$ and $q = q_L$ lie below the vigilance threshold. Hence

$e_i^*(0) = e_i^*(q_L) = 0$. Therefore,

$$D_i(q_L) - D_i(0) = \left(s_N^0 - \ell q_L\right) - s_N^0 = -\ell q_L < 0.$$

Because $q_H \geq q_i^*$, reviewer i chooses vigilant review under high prevalence injection. Hence

$$D_i(q_H) - D_i(0) = \left(s_N^0 + \Delta_N - \ell q_H\right) - s_N^0 = \Delta_N - \ell q_H.$$

This difference is positive if and only if $\Delta_N > \ell q_H$. □

Proposition 2 characterizes the two competing effects of error injection. At low injection levels, reviewers remain in the routine review regime and injected errors primarily consume review capacity. At higher injection levels, naturally occurring AI error detection improves when the detection gain from vigilance exceeds the attentional cost imposed by injected errors.

A.2 Heterogeneous Reviewers

Reviewer specific vigilance costs may differ across reviewers. Let c_i be distributed according to cumulative distribution function F, while the remaining primitives are common across reviewers. Reviewer i chooses vigilant review whenever

$$c_i \leq B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right].$$

The fraction of reviewers choosing vigilant review under injected error prevalence q is

$$V(q) = F\left(B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right]\right).$$

Average detection of naturally occurring AI errors is

$$\overline{D}(q) = s_N^0 - \ell q + \Delta_N V(q).$$

Proposition 3. *Suppose $\Delta_I > \Delta_z$ and F is weakly increasing. Then the fraction of reviewers choosing vigilant review is weakly increasing in injected error prevalence:*

$$V(q) = F\left(B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right]\right).$$

Average detection of naturally occurring AI errors is

$$\overline{D}(q) = s_N^0 - \ell q + \Delta_N F\left(B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right]\right).$$

Proof. Because $\Delta_I > \Delta_z$, the argument of F,

$$B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right],$$

is increasing in q. Since F is weakly increasing, $V(q)$ is weakly increasing. Substituting this expression into the detection function and averaging across reviewers yields

$$\overline{D}(q) = s_N^0 - \ell q + \Delta_N F\left(B\left[\Delta_z + p(\Delta_N - \Delta_z) + q(\Delta_I - \Delta_z)\right]\right).$$

□

Proposition 3 extends the individual reviewer model to a heterogeneous reviewer population. Heterogeneity smooths the threshold behavior in Proposition 1, producing a continuous relationship between injected error prevalence and average detection. At low injection levels, relatively few reviewers switch into vigilant review and the attentional cost term can dominate. As injected error prevalence increases, more reviewers become vigilant, increasing detection of naturally occurring AI errors.

A.3 Injected Errors as a Measure of Reviewer Vigilance

Injected errors have known ground truth and therefore make reviewer performance directly observable. We characterize the relationship between reviewer performance on injected errors and reviewer performance on naturally occurring AI errors.

Fix an injection condition with $q > 0$. Let $M_{Ii}(q)$ denote reviewer i's injected error detection rate and let $M_{Ni}(q)$ denote reviewer i's naturally occurring AI error detection rate. Using the detection functions defined above,

$$M_{Ii}(q) = s_I(e_i^*(q)) + \varepsilon_{Ii} = s_I^0 + \Delta_I e_i^*(q) + \varepsilon_{Ii},$$

and

$$M_{Ni}(q) = s_N(e_i^*(q)) - \ell q + \varepsilon_{Ni} = s_N^0 + \Delta_N e_i^*(q) - \ell q + \varepsilon_{Ni}.$$

The noise terms capture idiosyncratic variation in reviewer level detection rates. Assume they are mean zero, mutually uncorrelated, and uncorrelated with $e_i^*(q)$.

Proposition 4. *Fix injected error prevalence $q > 0$. Then*

$$\operatorname{Cov}\left(M_{Ii}(q), M_{Ni}(q)\right) = \Delta_I \Delta_N \operatorname{Var}\left(e_i^*(q)\right).$$

Proof. Since q is fixed, the terms s_I^0, s_N^0, and $-\ell q$ are constant across reviewers. Therefore,

$$\begin{aligned}\operatorname{Cov}\left(M_{Ii}(q), M_{Ni}(q)\right) &= \operatorname{Cov}\left(s_I^0 + \Delta_I e_i^*(q) + \varepsilon_{Ii}, s_N^0 + \Delta_N e_i^*(q) - \ell q + \varepsilon_{Ni}\right) \\ &= \Delta_I \Delta_N \operatorname{Var}\left(e_i^*(q)\right),\end{aligned}$$

where the second equality follows because constants do not affect covariance and the noise terms are uncorrelated with $e_i^*(q)$ and with each other. □

Proposition 4 characterizes the reviewer level relationship between injected error detection and naturally occurring AI error detection. The covariance increases with the sensitivity of injected error detection to vigilance, the sensitivity of naturally occurring AI error detection to vigilance, and cross reviewer variation in vigilance. When vigilance improves detection of both error types and reviewers differ in vigilance within an injection condition, performance on injected errors provides an observable signal of reviewer vigilance.

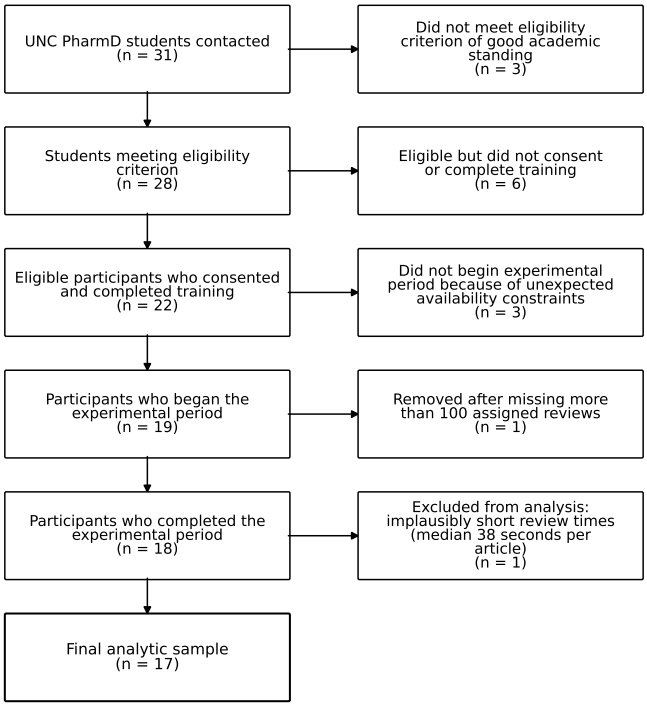

Figure 10: Participant recruitment, study participation, and exclusions from the final analytic sample.

A.4 Participant Flow Diagram

B Standardized Review Manual

Listing 1: Standardized manual used for summary generation and error identification.

```
Manual for Error Identification

1. Age

Correct entry:
Enter the age of the patient as reported in the article. Use the most detailed category as
    reported in the article.

Example:
"42 years", "patient in their 40s", "adult"

Errors include:
- Omitting age when it is explicitly stated.
- Entering an incorrect or estimated age without evidence.
- Using vague terms such as "adult" or "middle-aged" instead of a numeric value when a more
    specific value is provided.

2. Gender
```

```
Correct entry:
Enter the gender of the patient as stated or clearly implied in the article (e.g., through
    pronouns).

Example:
"Male", "Female", "Transgender Male", "Transgender Female", "Non-binary"

Errors include:
- Guessing gender when not specified.
- Omitting or writing "unknown" when the article clearly indicates gender.

3. Historical drug

Correct entry:
List all drugs that were stopped before the exposure period leading to the adverse event. These
    are medications that the patient was no longer taking by the time the suspect drug was being
    taken. Include duration, indication, dosage, and start/stop dates when available.

Note:
There is no need to evaluate whether the drug might still have been active in the patient's system
    (e.g., long half-life). Classification is based strictly on calendar timing of use, not
    pharmacologic activity.

Errors include:
- Including medications that the patient was taking during the period the suspect drug was taken.
- Including drugs that were started after the onset of the adverse event.
- Omitting indications when they are provided.
- Inferring indications that are not explicitly stated.

4. Medical history

Correct entry:
List all pre-existing conditions, chronic illnesses, or comorbidities that occurred before the
    adverse event. Include substance use history if stated.

Errors include:
- Including the adverse event itself as part of medical history.
- Listing medications instead of diagnoses.
- Omitting medical history mentioned in the article.

5. Suspect drug(s)

Correct entry:
Enter the drug(s) identified by the authors as the likely cause of the adverse event, using the
    same name (generic and/or brand; may include drug class if referenced).
- If an article states a drug "cannot be excluded" but does not treat it as a primary suspect, it
    should not be included as a suspect drug.
- If multiple drugs or adverse events are treated as primary, all of them should be included, and
    each suspect drug should be clearly matched to its corresponding adverse event.
- If the adverse event is attributed to a drug interaction, all drugs involved in the interaction
    should be included

Errors include:
```

```
- Omitting the brand name when it is part of the reported data.
- Substituting a drug inferred from context but not explicitly identified.

6. Indication

Correct entry:
Enter the reason the suspect drug was prescribed or taken. This may include off-label use or
    illicit use if stated. Use only the information provided in the article and do not rely on
    outside knowledge or assumptions.

Errors include:
- Recording the adverse event itself as the indication.
- Omitting the indication when clearly stated.
- Confusing the indication with mechanism of action or clinical outcome.
- Including or specifying an indication that is not explicitly stated in the article. For example,
    stating "ulcerative colitis" when the article only mentions "inflammatory bowel disease."

7. Dosing regimen

Correct entry:
Include the dose, frequency, duration, and route as stated (e.g., "oral, 25 mg twice daily for 12
    days"). Use only the information provided in the article and do not rely on outside knowledge
    or assumptions.

Errors include:
- Assuming a route of administration that is not explicitly stated in the article.
- Assuming a frequency that is not explicitly stated in the article (e.g., listing "25 mg once
    daily" when the article states "25 mg/day").
- Providing only partial information when more details are available (e.g., listing only "oral 25
    mg").
- Combining multiple dosing regimens without distinguishing them.

8. Start date of suspect drug

Correct entry:
Record the calendar or relative date when the suspect drug was initiated.

Errors include:
- Entering the date of event onset instead.
- Using admission date or diagnosis date as start date.
- Omitting the start date.

9. Stop date of suspect drug

Correct entry:
Enter the date or relative timing when the suspect drug was discontinued. If not discontinued,
    report that it was continued.

Errors include:
- Using event onset date if discontinuation occurred earlier.
- Writing "stopped" without specifying timing.
- Indicating the drug was stopped when it was not.
- Omitting the stop date or duration.
```

```
10. Disposition of drug

Correct entry:
Describe the action taken regarding the suspect drug (e.g., "stopped," "dose reduced," or
    "continued").

Errors include:
- Reporting outcomes instead of actions.
- Omitting the action when stated.
- Providing the wrong disposition.

11. Concomitant medications

Correct entry:
List medications that the patient was actively taking during the same time period as the suspect
    drug. Do NOT classify any medication started after the onset of the adverse event as
    concomitant (unless multiple, separate adverse events are described). Medications fully
    discontinued before the suspect drug was started are not concomitant and should be listed as
    historical if relevant.

Errors include:
- Including medications started after the adverse event onset.
- Including medications discontinued before the suspect drug was started.
- Listing medications not mentioned in the article.
- Failing to list concomitant medications when described.
- Misspelling drug names.

12. Event(s)

Correct entry:
Enter all adverse events using the symptoms and/or diagnoses as described (e.g., "drug-induced
    liver injury," "cholestatic hepatitis").

Errors include:
- Making diagnoses or inferences not stated in the article.
- Listing only symptoms when a diagnostic term is provided.
- Omitting the event of interest.
- Providing the wrong event.

13. Event onset date

Correct entry:
Record the first calendar date or relative date of symptoms or laboratory abnormalities as
    reported (e.g., on 17 September, 4 days after starting the suspect drug, etc.). Event onset
    should be defined as the time when symptoms or laboratory abnormalities related to the adverse
    event first begin. If there is uncertainty about whether the symptoms or abnormalities are
    related to the adverse event, consider them related unless they are explicitly attributed to
    another cause.

Errors include:
- Recording the wrong event date.
- Omitting the onset date.
```

14. Time to onset

Correct entry:
State the interval between drug initiation (or withdrawal, if applicable) and event onset. Event onset should be defined as the time when symptoms or laboratory abnormalities related to the adverse event first begin. If there is uncertainty about whether the symptoms or abnormalities are related to the adverse event, consider them related unless they are explicitly attributed to another cause.

Errors include:
- Confusing with event duration.
- Omitting when present.
- Providing the wrong interval.

15. Treatment product

Correct entry:
List all treatments provided to manage the adverse event and describe their outcome. All treatments relevant to the adverse event should be included, even if they occur across multiple episodes or admissions.

Errors include:
- Listing the suspect drug as treatment.
- Including therapy for unrelated conditions.
- Omitting treatments.

16. Outcome of adverse event

Correct entry:
Summarize the clinical outcome of the adverse event: resolved, improved, worsened, fatal, or unknown.

Errors include:
- Substituting dechallenge result for outcome.
- Omitting outcome when reported.
- Providing the wrong outcome.

17. Dechallenge

Correct entry:
Positive: The adverse event improved after the suspect drug was discontinued. Ignore the timing of improvement and whether other treatments were given.
Negative: The adverse event did not improve after the suspect drug was discontinued.
Unknown: The dechallenge was not assessed or there is insufficient information to determine the outcome.

Errors include:
- Assigning the wrong dechallenge category.

18. Rechallenge

Correct entry:

```
Positive: The adverse event recurred after systemic re-exposure to the suspect drug (e.g., oral,
    IV, IM rechallenge, including a supervised oral challenge).
Negative: The adverse event did not recur after systemic re-exposure to the suspect drug.
Unknown: There was no systemic re-exposure to the drug, or rechallenge information is not reported
    or unclear.

Key clarification:
A rechallenge requires systemic re-administration of the drug capable of reproducing the original
    adverse event.
Oral challenge tests count as rechallenge, because they involve clinically meaningful systemic
    re-exposure.
Patch tests and other localized or in vitro tests do not count as rechallenge, because they do not
    recreate systemic exposure or the original clinical reaction, and instead provide only
    supportive diagnostic or immunologic evidence.

Errors include:
- Confusing unrelated symptoms or illnesses with a true recurrence of the adverse event after drug
    re-exposure.
- Treating a patch test, skin test, lymphocyte transformation test, or other diagnostic assay as a
    positive rechallenge.
- Labeling rechallenge as positive when the drug was not actually re-administered systemically.
```

C Taxonomy of AI Failure Modes

Table 6: Taxonomy of AI failure modes in adverse drug event report summarization

Error category	**Description**	**Illustrative example**
Unsupported inference	The summary introduces details that are not stated in the source report. This may include assuming the route of administration, dose, administration frequency, indication, drug disposition, treatment, time to onset, diagnosis, patient characteristics, or other unstated details.	The report states that the patient was treated for hyperthyroidism, but the summary states that the patient has "Graves' hyperthyroidism."
Omission or incomplete extraction	The summary fails to include clinically relevant information from the report. This may include omitting a suspect drug, an important drug interaction, a historical drug, a concomitant medication, a relevant medical condition, a treatment product, rechallenge information, an adverse event, a distinct occurrence of an event, or key details that materially affect interpretation of the adverse event.	The report identifies two interacting suspect drugs, but the summary lists only one.

Continued on next page

Table 6 – continued from previous page

Error category	Description	Illustrative example
Incorrect field classification	The summary extracts information from the report but assigns it to the wrong field. This may include classifying a historical drug as a concomitant medication, a concomitant medication as a historical drug, a suspect drug as a concomitant medication, a treatment product as a concomitant medication, or an adverse event as medical history.	The patient received tacrolimus to treat the adverse event after the suspect drug was discontinued, but the summary lists tacrolimus as a concomitant medication.
Incorrect extraction of reported information	The summary inaccurately records information from the report. This may include errors in patient age, drug identity, formulation, dose, frequency, route, units, duration, indication, dates, or other reported values. Unlike unsupported inference, the relevant information is present in the source report but is extracted incorrectly.	The report states that the patient received two injections of insulin glargine, but the summary states that the dosing regimen consisted of a single injection.
Temporal reasoning error	The summary misrepresents the sequence or timing of exposures, symptoms, treatments, or outcomes. This may include using an incorrect start date or stop date, calculating time to onset incorrectly, failing to use the first symptom onset, misstating the duration of treatment or symptoms, or merging separate dosing regimens or event episodes into a single timeline.	The patient developed a rash three days after drug initiation and respiratory symptoms one week later, but the summary calculates time to onset using the later respiratory symptoms.
Cross-context contamination	The summary incorporates information that appears in the article but does not apply to the patient or event being summarized. This may include using details from other cases in the article, incorporating general background information, or conflating details about separate events.	The report notes that the suspect drug is known to interact with several medications, but the summary incorrectly identifies those medications as drugs taken by the patient.

Continued on next page

Table 6 – continued from previous page

Error category	Description	Illustrative example
Causality or outcome interpretation error	The summary incorrectly interprets the patient's clinical course or the relationship between the suspect drug and the adverse event. This may include misclassifying dechallenge or rechallenge as positive, negative, or unknown, omitting a documented rechallenge, or overstating or understating the event outcome.	The patient's symptoms ultimately improved after the suspect drug was discontinued, but the summary records a negative dechallenge.

D Distribution of true and injected errors

Figure 11 compares where true errors occurred in the original AI-generated summaries with where errors were injected under each error-injection condition. True errors are counted at the article-variable level across the 600 experimental articles. Injected errors are counted separately using the low-, medium-, and high-injection versions of the 600 articles.

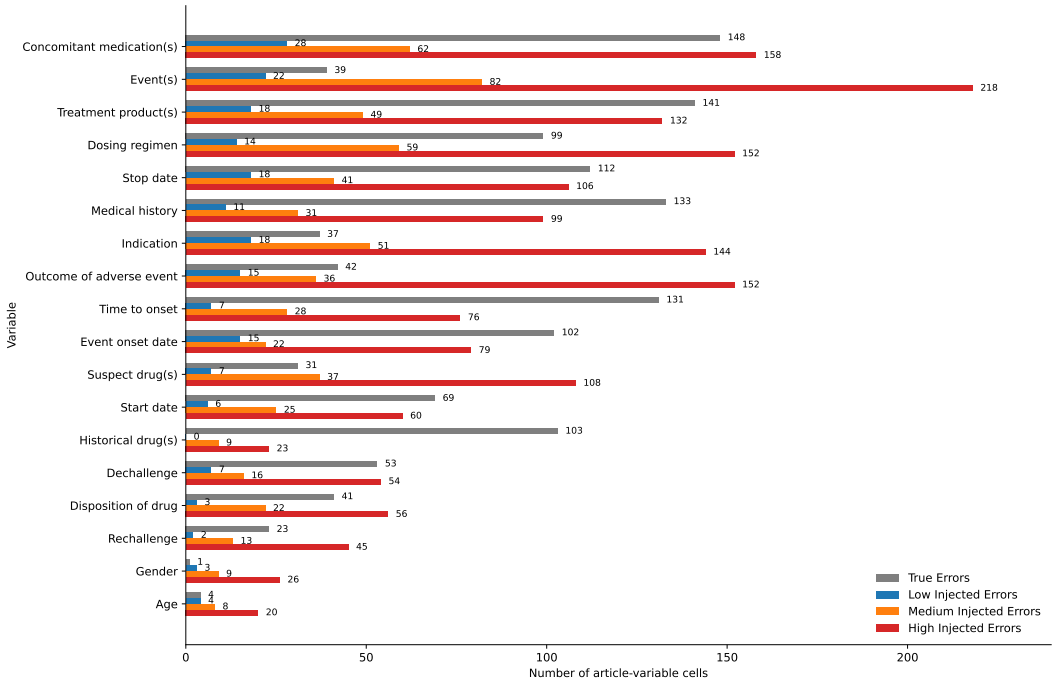

Figure 11: Frequency of true errors and injected errors by summary variable.

We used chi-square tests of homogeneity to assess whether true errors and injected errors were distributed similarly across summary variables. The omnibus test compared naturally occurring true errors with low-, medium-, and high-frequency injected errors. Pairwise tests compared true errors with each injected-error condition, with Benjamini–Hochberg adjustment across the three pairwise comparisons.

Appendix Table 7 shows that true and injected errors differed significantly in their distribution across variables. The omnibus test rejected equality of the variable-level distributions ($\chi^2(51) = 456.99$, $p < 0.001$, Cramér's $V = 0.20$). Pairwise comparisons likewise showed significant differences between true errors and each injected-error condition. True errors were relatively more concentrated in historical drugs, medical history, time to onset, event onset date, and treatment products, whereas injected errors were relatively more concentrated in events, event outcome, indication, and suspect drugs.

These differences partly reflect the injection design. To preserve identifiable ground truth, injected errors were generally not placed in cells that already contained naturally occurring true errors. As a result, fields with higher rates of true errors had fewer eligible clean cells available for error injection, creating a tradeoff between preserving true-error detection opportunities and matching the variable-level distribution of naturally occurring model failures.

Table 7: Tests comparing the variable-level distribution of true and injected errors

Comparison	χ^2	**df**	p**-value**	**Cramér's** V
Omnibus: true, low, medium, and high	456.99	51	< 0.001	0.20
True errors vs. low injected errors	112.13	17	< 0.001	0.27
True errors vs. medium injected errors	226.62	17	< 0.001	0.34
True errors vs. high injected errors	375.94	17	< 0.001	0.35

Note: Tests are chi-square tests of homogeneity comparing the distribution of errors across summary variables. The omnibus test compares naturally occurring true errors with low-, medium-, and high-frequency injected errors. Pairwise p-values are Benjamini–Hochberg adjusted across the three true-versus-injected comparisons.

We also tested whether the injected-error distributions differed across the low-, medium-, and high-injection conditions. Appendix Table 8 shows no statistically significant differences in the variable-level distribution of injected errors across injection conditions. The omnibus test did not reject equality of the low-, medium-, and high-injection distributions ($\chi^2(34) = 31.42$, $p = 0.595$, Cramér's $V = 0.08$). Pairwise comparisons were likewise not statistically significant after Benjamini–Hochberg adjustment. Together, these findings indicate that although the variable distribution of injected errors differed from that of naturally occurring true errors, the injected-error distribution itself remained consistent across the low-, medium-, and high-frequency injection conditions.

Table 8: Tests comparing the variable-level distribution of injected errors across conditions

Comparison	χ^2	**df**	p**-value**	**Cramér's** V
Omnibus: low, medium, and high injected errors	31.42	34	0.595	0.08
Low vs. medium injected errors	20.55	17	0.370	0.16
Low vs. high injected errors	21.89	17	0.370	0.11
Medium vs. high injected errors	9.16	17	0.935	0.06

Note: Tests are chi-square tests of homogeneity comparing the distribution of injected errors across summary variables in the low-, medium-, and high-injection conditions. The omnibus p-value is unadjusted. Pairwise p-values are Benjamini–Hochberg adjusted across the three pairwise condition comparisons.

E Balance of error-opportunity proportions across randomized sequences

As a descriptive analysis, we examined whether true-error and injected-error opportunities were evenly distributed across randomized sequences within each injected-error condition. For each condition-by-sequence cell, we calculated the proportion of variable cells that contained a true-error opportunity or an injected-error opportunity. We then tested whether these proportions differed across randomized sequences within each condition.

Appendix Table 9 shows the proportion of true-error and injected-error opportunities by condition and randomized sequence. Injected-error opportunity proportions were similar across sequences within each injected-error condition. In the low-injection condition, injected-error opportunity proportions ranged from 1.78% to 1.94%; in the medium-injection condition, from 5.54% to 5.67%; and in the high-injection condition, from 15.10% to 16.12%. Omnibus tests did not reject equality of injected-error opportunity proportions across sequences in the low-, medium-, or high-injection conditions (Appendix Table 10).

True-error opportunity proportions were also similar across sequences for the no-injection, medium-injection, and high-injection conditions. The only statistically significant sequence difference was observed for true-error opportunities in the low-injection condition ($p < 0.001$), where one sequence had a slightly lower proportion of naturally occurring true-error opportunities. Because the primary true-error detection models condition on true-error opportunities and adjust for reviewer, study period, document, and variable fixed effects, this imbalance is unlikely to explain the main detection results.

Table 9: Proportion of true-error and injected-error opportunities by condition and randomized sequence

Condition	Sequence	Cells	True-error %	Injected-error %
None	S1	10,800	11.9	0.0
None	S2	13,194	12.2	0.0
None	S3	13,500	12.6	0.0
None	S4	8,100	12.0	0.0
Low	S1	10,800	11.8	1.9
Low	S2	13,500	11.7	1.9
Low	S3	12,834	11.9	1.8
Low	S4	8,100	11.0	1.9
Medium	S1	10,800	11.4	5.6
Medium	S2	13,500	11.4	5.6
Medium	S3	13,500	11.0	5.5
Medium	S4	8,100	12.6	5.7
High	S1	10,800	10.4	16.1
High	S2	13,140	10.2	16.1
High	S3	12,816	9.9	15.9
High	S4	8,100	10.0	15.1

Table 10: Tests for sequence differences in error-opportunity proportions within condition

Condition	True-error p-value	Injected-error p-value
None	0.239	
Low	<0.001	0.752
Medium	0.058	0.982
High	0.502	0.439

Note: P-values are Benjamini–Hochberg-adjusted omnibus Wald tests from logistic models of the error-opportunity indicator on randomized sequence, estimated separately within each condition with reviewer-clustered standard errors.

F Baseline balance before and after participant dropout and exclusion

Appendix Table 11 compares baseline characteristics across randomized sequences before participant dropout and analytic exclusion and in the final analytic sample. Before dropout and exclusion, the sample contained 19 participants, with sequence counts of 5, 5, 5, and 4. After excluding two additional participants, the final analytic sample contained 17 participants, with sequence counts of 4, 5, 5, and 3. Baseline characteristics remained similar across randomized sequences in both samples. All one-way ANOVA p-values were greater than 0.70 before dropout and exclusion and greater than 0.44 in the final analytic sample, suggesting that participant dropout and exclusion did not meaningfully alter baseline balance across sequences.

Table 11: Baseline characteristics by randomized sequence before and after participant dropout and exclusion

Characteristic	Overall	S1	S2	S3	S4	p-value
Panel A. Before participant dropout and analytic exclusion						
Participants, n	19	5	5	5	4	
Age	24.89 (1.24)	24.60 (1.67)	25.40 (1.14)	25.00 (1.00)	24.50 (1.29)	0.707
Year in PharmD program	3.42 (0.90)	3.60 (0.55)	3.40 (1.14)	3.40 (0.55)	3.25 (1.50)	0.960
Familiarity with reviewing adverse drug event reports	2.68 (0.82)	2.80 (1.10)	2.40 (1.14)	2.80 (0.45)	2.75 (0.50)	0.867
Frequency of LLM use	3.58 (1.12)	3.40 (1.14)	4.00 (0.71)	3.60 (1.34)	3.25 (1.50)	0.790
Familiarity with LLMs	3.00 (0.67)	3.00 (0.00)	2.80 (0.45)	3.20 (0.45)	3.00 (1.41)	0.851
General trust in AI-system outputs	3.05 (1.03)	3.00 (1.41)	3.00 (1.00)	2.80 (0.84)	3.50 (1.00)	0.809
Trust in AI-generated adverse drug event summaries	3.05 (0.78)	3.00 (1.00)	3.40 (0.55)	2.80 (0.84)	3.00 (0.82)	0.705
Baseline attention-related cognitive errors	2.62 (0.49)	2.57 (0.39)	2.80 (0.65)	2.62 (0.65)	2.48 (0.08)	0.810
Panel B. Final analytic sample after participant dropout and exclusion						
Participants, n	17	4	5	5	3	
Age	24.88 (1.32)	24.50 (1.91)	25.40 (1.14)	25.00 (1.00)	24.33 (1.53)	0.686
Year in PharmD program	3.47 (0.87)	3.50 (0.58)	3.40 (1.14)	3.40 (0.55)	3.67 (1.53)	0.980
Familiarity with reviewing adverse drug event reports	2.76 (0.75)	3.25 (0.50)	2.40 (1.14)	2.80 (0.45)	2.67 (0.58)	0.441
Frequency of LLM use	3.65 (1.11)	3.75 (0.96)	4.00 (0.71)	3.60 (1.34)	3.00 (1.73)	0.711
Familiarity with LLMs	2.94 (0.66)	3.00 (0.00)	2.80 (0.45)	3.20 (0.45)	2.67 (1.53)	0.711
General trust in AI-system outputs	3.12 (0.93)	3.50 (1.00)	3.00 (1.00)	2.80 (0.84)	3.33 (1.15)	0.721
Trust in AI-generated adverse drug event summaries	3.00 (0.79)	3.00 (1.15)	3.40 (0.55)	2.80 (0.84)	2.67 (0.58)	0.587
Baseline attention-related cognitive errors	2.60 (0.49)	2.42 (0.24)	2.80 (0.65)	2.62 (0.65)	2.50 (0.08)	0.722

Note: Values are mean (standard deviation) unless otherwise indicated. p-values are from one-way analyses of variance across randomized sequences.

G PubMed Search Strategy

We identified publicly available adverse drug event case reports using the following PubMed search strategy:

Listing 2: PubMed search strategy used to identify adverse drug event case reports.

```
(
("Case Reports"[Publication Type])
AND Humans[Mesh]
AND (
"Drug-Related Side Effects and Adverse Reactions"[Mesh]
OR "Adverse Drug Reaction Reporting Systems"[Mesh]
OR "Pharmacovigilance"[Mesh]
OR "adverse drug reaction"[tiab]
OR "adverse drug event"[tiab]
OR "drug toxicity"[tiab]
OR "drug side effect"[tiab]
OR "drug-induced"[tiab]
OR "drug safety"[tiab]
OR "medication safety"[tiab]
OR "medication error"[tiab]
OR "treatment-emergent adverse event"[tiab]
OR "adverse event reporting system"[tiab]
OR "drug safety monitoring"[tiab]
OR "postmarketing surveillance"[tiab]
OR "spontaneous reporting"[tiab]
OR "pharmacovigilance system"[tiab]
)
AND (
"Cardiovascular Diseases"[Mesh]
OR "Heart Diseases"[Mesh]
OR "Hypertension"[Mesh]
OR "Arrhythmias, Cardiac"[Mesh]
OR "Heart Failure"[Mesh]
OR "Myocardial Ischemia"[Mesh]

‘‘‘
OR "Diabetes Mellitus"[Mesh]
OR "Metabolic Diseases"[Mesh]
OR "Insulin Resistance"[Mesh]
OR "Hypoglycemic Agents"[Mesh]

OR "Endocrine System Diseases"[Mesh]
OR "Hormones"[Mesh]
OR "Thyroid Diseases"[Mesh]
OR "Pituitary Diseases"[Mesh]
OR "Adrenal Gland Diseases"[Mesh]

OR "Eye Diseases"[Mesh]
OR "Ophthalmology"[Mesh]
OR "Retinal Diseases"[Mesh]
OR "Macular Degeneration"[Mesh]
OR "Glaucoma"[Mesh]
```

```
OR "Corneal Diseases"[Mesh]

OR "Substance-Related Disorders/drug therapy"[Mesh]
OR "Opioid-Related Disorders/drug therapy"[Mesh]
OR "Alcohol-Related Disorders/drug therapy"[Mesh]
OR methadone[tiab]
OR buprenorphine[tiab]
OR naltrexone[tiab]
OR naloxone[tiab]
OR disulfiram[tiab]
OR acamprosate[tiab]
OR varenicline[tiab]
OR bupropion[tiab]

OR "Dermatologic Agents"[Mesh]
OR isotretinoin[tiab]
OR retinoid*[tiab]
OR corticosteroid*[tiab]
OR "topical steroid*"[tiab]
OR dupilumab[tiab]
OR adalimumab[tiab]
OR biologic*[tiab]

OR "Contraceptive Agents"[Mesh]
OR contracept*[tiab]
OR "oral contraceptive*"[tiab]
OR levonorgestrel[tiab]
OR "ethinyl estradiol"[tiab]
OR progesterone[tiab]
OR intrauterine[tiab]

OR "Anti-Inflammatory Agents, Non-Steroidal"[Mesh]
OR NSAID*[tiab]
OR ibuprofen[tiab]
OR naproxen[tiab]
OR diclofenac[tiab]
OR corticosteroid*[tiab]
OR prednisone[tiab]
OR methotrexate[tiab]
OR biologic*[tiab]
OR DMARD*[tiab]

OR "Antipsychotic Agents"[Mesh]
OR antipsychotic*[tiab]
OR clozapine[tiab]
OR olanzapine[tiab]
OR risperidone[tiab]
OR quetiapine[tiab]
OR aripiprazole[tiab]
OR haloperidol[tiab]

OR "Analgesics"[Mesh]
OR analgesic*[tiab]
```

```
OR opioid*[tiab]
OR morphine[tiab]
OR oxycodone[tiab]
OR hydrocodone[tiab]
OR fentanyl[tiab]
OR tramadol[tiab]
OR gabapentin[tiab]
OR pregabalin[tiab]
‘‘‘

)
NOT (
"Review"[Publication Type]
OR "Systematic Review"[Publication Type]
OR "Meta-Analysis"[Publication Type]
OR "case series"[tiab]
)
AND free full text[sb]
AND english[lang]
)
```

H Prompting Pipeline

Listing 3: Prompt used to generate structured AI summaries of adverse drug event case reports.

```
Answer the requested fields using only the information provided in the case report. Follow the
    standardized review manual when extracting and classifying information. Do not rely on outside
    knowledge or make unsupported assumptions.

============================================================
STANDARDIZED REVIEW MANUAL
============================================================
{review_manual}

============================================================
CASE REPORT
============================================================
{case_text}

============================================================
TASK
============================================================

Complete all 18 fields defined in the standardized review manual.

Requirements:
- Answer all fields clearly and systematically.
- Preserve the variable names and ordering specified in the manual.
- Use only information supported by the case report.
- Do not infer details that are not explicitly stated or clearly indicated.
- Include all relevant information requested for each variable when available.
```

```
- If information is not reported or cannot be determined from the case report, state that it is
    unknown.
```

Listing 4: Prompt used to generate realistic injected errors in AI-generated adverse drug event summaries.

```
You are given an ARTICLE and a structured SUMMARY containing variables 1--18. Your task is to
    inject exactly {n_errors} new content errors into the SUMMARY while preserving all other text
    exactly as written.

The purpose of this task is to simulate realistic errors that may occur when an AI system
    summarizes an adverse drug event report.

============================================================
ARTICLE
============================================================
{article}

============================================================
ORIGINAL STRUCTURED SUMMARY
============================================================
{summary}

============================================================
REVIEW MANUAL FOR VARIABLES 1--18
============================================================
{review_manual}

============================================================
REALISTIC ERROR TAXONOMY
============================================================
{error_taxonomy}

============================================================
TASK INSTRUCTIONS
============================================================

STEP 1 -- IDENTIFY PRE-EXISTING ERRORS INTERNALLY

Compare each variable in the ORIGINAL STRUCTURED SUMMARY with the ARTICLE using the REVIEW MANUAL.

Identify which variables already contain genuine content errors. Use this information only to
    guide error injection.

Do NOT report or correct pre-existing errors.

Ignore minor grammar, spelling, stylistic, and formatting issues unless they change the clinical
    meaning.

STEP 2 -- INJECT EXACTLY {n_errors} NEW CONTENT ERRORS

Modify exactly {n_errors} different variables in the ORIGINAL STRUCTURED SUMMARY.
```

```
Each selected variable must have been correct before modification.

Each injected error must be:
- clinically meaningful;
- clearly incorrect when compared with the ARTICLE;
- realistic for an AI-generated summary;
- unambiguous and independently verifiable;
- limited to a single variable.

Use the REALISTIC ERROR TAXONOMY to guide error injection. Across summaries, diversify both the
    variables modified and the categories of errors introduced.

Do NOT:
- modify a variable that already contains a pre-existing content error;
- introduce more than one injected error into the same variable;
- alter variables that are not selected for error injection;
- correct any pre-existing errors;
- rewrite, reformat, or paraphrase text unnecessarily;
- add commentary inside the modified summary;
- inject an error that depends on outside medical knowledge rather than the ARTICLE;
- inject an error that could reasonably be interpreted as correct;
- inject an error solely by introducing a minor typographical or grammatical mistake.

Preserve all text in unmodified variables exactly as written in the ORIGINAL STRUCTURED SUMMARY.

If it is not possible to inject exactly {n_errors} clear and indisputable errors without modifying
    a variable that already contains an error, inject the maximum feasible number of valid new
    errors.

============================================================
STRICT OUTPUT FORMAT
============================================================

Return valid JSON only. Do not include markdown, code fences, or any additional commentary.

Use exactly the following structure:

{
  "modified_summary": "FULL SUMMARY TEXT AFTER YOUR CHANGES",
  "error_explanations": [
    "Attention: Variable X (\"VARIABLE NAME\") contains an error in the summary. BRIEF EXPLANATION
    OF THE ERROR",
    "Attention: Variable X (\"VARIABLE NAME\") contains an error in the summary. BRIEF EXPLANATION
    OF THE ERROR"
  ]
}

Requirements:

1. "modified_summary"
- Return the complete SUMMARY after injecting the new errors.
- Preserve the original formatting of the SUMMARY.
- Preserve the original wording of all unmodified variables exactly.
```

```
- Do not mark, highlight, or otherwise identify the injected errors within the SUMMARY itself.

2. "error_explanations"
- Return an array containing exactly {n_errors} strings.
- Include one string for each newly injected error.
- Each string must follow exactly this format:

  Attention: Variable X ("VARIABLE NAME") contains an error in the summary. <brief explanation of
    the error>

- X must be the correct variable number for the modified variable.
- "VARIABLE NAME" must exactly match the variable name as written in the SUMMARY.
- Each explanation must describe the specific error introduced into the final SUMMARY.
- Each explanation must be brief and understandable without reference to internal reasoning.
- All X values must be different.
- Describe only the newly injected errors.
- Do not mention any pre-existing errors.
- Do not include evidence quotations, audit notes, or explanations of variables that were not
    modified.

Examples:

Attention: Variable 3 ("Historical drugs") contains an error in the summary. The summary
    incorrectly lists tacrolimus as a historical drug even though the article states that it was
    administered after the adverse event developed.

Attention: Variable 4 ("Medical history") contains an error in the summary. The summary omits
    chronic kidney disease, which is described in the article as part of the patient's medical
    history.

Attention: Variable 14 ("Time to onset") contains an error in the summary. The summary states that
    the event began 10 days after treatment initiation, but the article reports that the first
    symptoms appeared after 3 days.

Before returning the JSON, verify internally that:
- the JSON is valid;
- exactly {n_errors} variables were modified;
- the "error_explanations" array contains exactly {n_errors} strings;
- each explanation corresponds to one modified variable;
- no variable number appears more than once;
- all unmodified variables remain unchanged;
- no pre-existing errors are described in "error_explanations".

Do not include any output other than the JSON object.
```

I Survey Instruments

I.0.1 Pre-Experiment Survey

Background Characteristics

1. What is your age?

Response format: Dropdown menu.

2. What year are you in your PharmD program?

- ○ P1
- ○ P2
- ○ P3
- ○ P4
- ○ Other: _____

3. How would you rate your familiarity with large language models (e.g., ChatGPT, Claude, Gemini)?

- ○ Not at all familiar
- ○ Slightly familiar
- ○ Moderately familiar
- ○ Very familiar
- ○ Extremely familiar

4. How often do you use large language models (e.g., ChatGPT, Claude, Gemini)?

- ○ Never
- ○ Rarely (once a month or less)
- ○ Occasionally (a few times a month)
- ○ Frequently (a few times a week)
- ○ Daily

5. How familiar are you with reviewing published articles or case reports that describe an adverse event related to a drug?

- ○ Not at all familiar — I have never reviewed these types of reports and am not familiar with their structure or content.
- ○ Slightly familiar — I have encountered these reports before but have rarely reviewed them in detail.
- ○ Moderately familiar — I have reviewed some of these reports and understand their typical structure and purpose.
- ○ Very familiar — I have reviewed many such reports and feel comfortable evaluating their clinical content.
- ○ Extremely familiar — I routinely review these reports and consider myself highly experienced in interpreting them.

6. In what contexts have you used AI or large language models before? Select all that apply.

- ☐ Academic writing
- ☐ Clinical education
- ☐ Research
- ☐ Entertainment or personal use
- ☐ I have never used one
- ☐ Other: _____

Trust in AI Systems

Please indicate how much you agree or disagree with the following statements.
Response scale: 1 = Strongly disagree; 5 = Strongly agree.

1. I generally trust the output of AI systems.
2. To ensure you are paying attention, please select "Strongly Agree" for this statement.
3. I can trust AI to generate accurate summaries of adverse drug event reports.

Everyday Attention

Please answer each question using the scale below.
Response scale: 1 = Never; 2 = Rarely; 3 = Sometimes; 4 = Often; 5 = Very often.

1. I have gone to the fridge to get one thing (e.g., milk) and taken something else (e.g., juice).
2. I go into a room to do one thing (e.g., brush my teeth) and end up doing something else (e.g., brush my hair).
3. I have lost track of a conversation because I zoned out when someone else was talking.
4. I have absent-mindedly placed things in unintended locations (e.g., putting milk in the pantry or sugar in the fridge).
5. I have gone into a room to get something, got distracted, and wondered what I went there for.
6. I begin one task and get distracted into doing something else.
7. When reading, I find that I have read several paragraphs without being able to recall what I read.
8. I make mistakes because I am doing one thing and thinking about another.
9. I have absent-mindedly mixed up targets of my action (e.g., pouring or putting something into the wrong container).
10. I have to go back to check whether I have done something or not (e.g., turning out lights, locking doors).
11. I have absent-mindedly misplaced frequently used objects, such as keys, pens, or glasses.
12. I fail to see what I am looking for even though I am looking right at it.

I.0.2 Weekly Survey

Task Load

1. How mentally demanding were the tasks you completed during the past week?
Response scale: 1 = Not demanding at all; 7 = Extremely demanding.

2. How physically demanding were your tasks this past week?
Response scale: 1 = Not demanding at all; 7 = Extremely demanding.

3. How hurried or rushed did you feel while completing your tasks this past week?
Response scale: 1 = Not rushed at all; 7 = Extremely rushed.

4. How successful do you think you were in accomplishing the tasks you were given this week?
Response scale: 1 = Very poor; 7 = Excellent.

5. How hard did you have to work to achieve your level of performance during this week?
Response scale: 1 = Very low effort; 7 = Very high effort.

6. How irritated, stressed, or discouraged did you feel while completing your tasks this past week?
Response scale: 1 = Not at all; 7 = Extremely.

7. How high was your overall workload outside of the experiment this past week (e.g., school, work, clinical responsibilities)?
Response scale: 1 = Very low; 7 = Very high.

8. How confident did you feel in your ability to evaluate the summaries this week?
Response scale: 1 = Not confident at all; 7 = Extremely confident.

9. To what extent do you trust the AI-generated summaries you reviewed this week?
Response scale: 1 = Not at all; 7 = Completely.

10. How would you describe your attention while reviewing summaries this week?

1. I was highly attentive throughout.
2. I was mostly attentive, but lost focus at times.
3. I struggled to stay focused.
4. Other: _____

Trust in AI Systems

Please indicate how much you agree or disagree with the following statements.
Response scale: 1 = Strongly disagree; 5 = Strongly agree.

1. I generally trust the output of AI systems.
2. To help us verify data quality, please choose "Strongly Agree" for this question.
3. I can trust AI to generate accurate summaries of adverse drug event reports.
4. I can trust the AI model that generated the summaries I reviewed this week to generate accurate summaries of adverse drug event reports.

Additional Questions

11. Have you noticed anything unusual or surprising about the summaries you reviewed this week?

- ○ Yes
- ○ No

If yes, please describe briefly: _____

12. Did your approach to reviewing the AI-generated summaries change this week? If so, how?
Response format: Open response.

13. Do you have any suggestions or feedback so far about the task or interface?
Response format: Open response.

I.0.3 Final Survey (Post-Experiment)

Task Load

1. How mentally demanding were the tasks you completed during the past week?
Response scale: 1 = Not demanding at all; 7 = Extremely demanding.

2. How physically demanding were your tasks this past week?
Response scale: 1 = Not demanding at all; 7 = Extremely demanding.

3. How hurried or rushed did you feel while completing your tasks this past week?
Response scale: 1 = Not rushed at all; 7 = Extremely rushed.

4. How successful do you think you were in accomplishing the tasks you were given this week?
Response scale: 1 = Very poor; 7 = Excellent.

5. How hard did you have to work to achieve your level of performance during this week?
Response scale: 1 = Very low effort; 7 = Very high effort.

6. How irritated, stressed, or discouraged did you feel while completing your tasks this past week?
Response scale: 1 = Not at all; 7 = Extremely.

7. How high was your overall workload outside of the experiment this past week (e.g., school, work, clinical responsibilities)?

Response scale: 1 = Very low; 7 = Very high.

8. How confident did you feel in your ability to evaluate the summaries this week?
Response scale: 1 = Not confident at all; 7 = Extremely confident.

9. To help us verify data quality, please choose "Extremely confident" for this question.
Response scale: 1 = Not confident at all; 7 = Extremely confident.

10. To what extent do you trust the AI-generated summaries you reviewed this week?
Response scale: 1 = Not at all; 7 = Completely.

11. How would you describe your attention while reviewing summaries this week?

1. I was highly attentive throughout.
2. I was mostly attentive, but lost focus at times.
3. I struggled to stay focused.
4. Other: _____

Feedback on the Summaries and Review Process

12. Have you noticed anything unusual or surprising about the summaries?

- ○ Yes
- ○ No

If yes, please describe briefly: _____

13. Have you noticed any significant differences between the errors included in the "Some Missed Errors" messages and errors you identified that were not included in those messages?

- ○ Yes
- ○ No

If yes, please describe: _____

Trust in AI Systems

Please indicate how much you agree or disagree with the following statements.
Response scale: 1 = Strongly disagree; 5 = Strongly agree.

1. I generally trust the output of AI systems.
2. I can trust AI to generate accurate summaries of adverse drug event reports.

Additional Questions

14. Did your approach to reviewing the AI-generated summaries change over time? If so, how?
Response format: Open response.

15. Did your approach to reviewing the AI-generated summaries change between the different LLM models used each month? If so, how?
Response format: Open response.

16. Is there anything you would change about this study or the task design?
Response format: Open response.

17. Any additional comments or feedback?
Response format: Open response.

J Additional Results

J.1 Stepwise robustness of true-error detection models

Appendix Tables 12 and 13 report stepwise robustness analyses for the true-error detection model. Starting from a base specification with reviewer and study-period fixed effects, we sequentially add cumulative review order, document fixed effects, variable fixed effects, and baseline covariates interacted with review order. The estimated pattern is stable across specifications: high-frequency error injection consistently increases true-error detection, whereas low-frequency error injection is consistently below the no-injection condition.

Table 12: **Stepwise robustness of true-error detection estimates, base specifications**

	Dependent variable: Detection of a visible true error			
	(1)	(2)	(3)	(4)
Low injected errors	0.708^{***}	0.707^{***}	0.712^{**}	0.707^{**}
	[0.556, 0.901]	[0.555, 0.900]	[0.550, 0.922]	[0.543, 0.920]
Medium injected errors	1.009	1.005	1.017	1.019
	[0.763, 1.335]	[0.760, 1.328]	[0.771, 1.341]	[0.767, 1.353]
High injected errors	1.615^{***}	1.613^{***}	1.664^{***}	1.686^{***}
	[1.254, 2.081]	[1.253, 2.077]	[1.283, 2.158]	[1.283, 2.215]
Reviewer fixed effects	Yes	Yes	Yes	Yes
Study-period fixed effects	Yes	Yes	Yes	Yes
Cumulative review order	No	Yes	Yes	Yes
Document fixed effects	No	No	Yes	Yes
Variable fixed effects	No	No	No	Yes
Familiarity × review order	No	No	No	No
Trust × review order	No	No	No	No
Attention errors × review order	No	No	No	No
Observations	17,612	17,612	17,612	17,612
McFadden pseudo-R^2	0.039	0.039	0.117	0.137
Omnibus condition p-value	< 0.001	< 0.001	< 0.001	< 0.001

Notes: Odds ratios are reported with 95% confidence intervals in brackets. The no-injected-error condition is the reference group. Stars are based on Benjamini–Hochberg-adjusted p-values; standard errors are clustered by reviewer. $^{*}p < 0.10$; $^{**}p < 0.05$; $^{***}p < 0.01$.

Table 13: **Stepwise robustness of true-error detection estimates, baseline-covariate interactions**

	Dependent variable: Detection of a visible true error		
	(5)	(6)	(7)
Low injected errors	0.710^{**}	0.722^{**}	0.723^{**}
	[0.543, 0.927]	[0.559, 0.933]	[0.559, 0.934]
Medium injected errors	1.018	0.999	0.999
	[0.770, 1.348]	[0.770, 1.296]	[0.771, 1.295]
High injected errors	1.680^{***}	1.619^{***}	1.619^{***}
	[1.277, 2.210]	[1.265, 2.073]	[1.265, 2.072]
Reviewer fixed effects	Yes	Yes	Yes
Study-period fixed effects	Yes	Yes	Yes
Cumulative review order	Yes	Yes	Yes
Document fixed effects	Yes	Yes	Yes
Variable fixed effects	Yes	Yes	Yes
Familiarity × review order	Yes	Yes	Yes
Trust × review order	No	Yes	Yes
Attention errors × review order	No	No	Yes
Observations	17,612	17,612	17,612
McFadden pseudo-R^2	0.137	0.138	0.138
Omnibus condition p-value	< 0.001	< 0.001	< 0.001

Notes: Odds ratios are reported with 95% confidence intervals in brackets. The no-injected-error condition is the reference group. Stars are based on Benjamini–Hochberg-adjusted p-values; standard errors are clustered by reviewer. $^{*}p < 0.10$; $^{**}p < 0.05$; $^{***}p < 0.01$.

J.2 True-error detection among subset of variables

Not all summary variables have the same clinical relevance across adverse-event reports. Some fields, such as time to onset, concomitant medications, medical history/comorbidities, rechallenge, and dechallenge, are broadly important for assessing the temporal, clinical, and causal relationship between a suspect drug and an adverse event. Other fields may be important in particular cases but are more episodic, depending on what information is reported and how it affects interpretation of the case. We therefore conducted a supplemental analysis restricted to these five broadly important variables (Appendix Figure 12).

Raw detection rates followed the same pattern as in the main analysis. Reviewers detected 26.9% of true errors in the no-injection condition, 23.8% in the low-injection condition, 27.8% in the medium-injection condition, and 33.2% in the high-injection condition. Detection varied across variables, ranging from 18.6% for medical history/comorbidities to 38.5% for concomitant medications.

In the fully adjusted model, true-error detection differed significantly across injected-error conditions (omnibus $p < 0.001$). Relative to the no-injection condition, high-frequency error injection significantly increased the odds of detecting true errors in these clinically important variables (OR, 1.45; 95% CI, 1.22–1.73; Benjamini–Hochberg-adjusted $p < 0.001$). Detection did not differ significantly in the low-injection condition (OR, 0.87; 95% CI, 0.70–1.09; adjusted $p = 0.341$) or the medium-injection condition (OR, 1.11; 95% CI, 0.86–1.42; adjusted $p = 0.423$). These results suggest that the main findings are not driven only by less clinically central fields; high-frequency error injection also improved detection among variables that are broadly important for pharmacovigilance review.

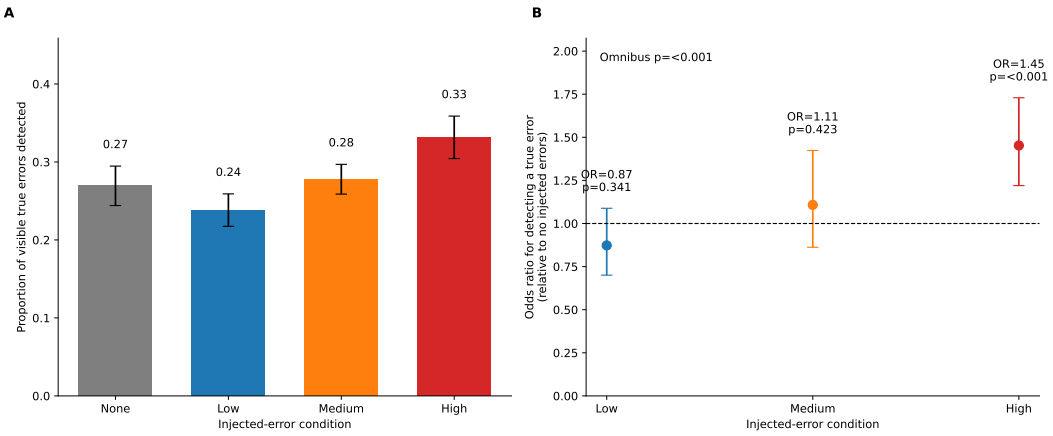

Figure 12: True-error detection among clinically important summary variables. The analysis is restricted to time to onset, concomitant medications, medical history/comorbidities, rechallenge, and dechallenge. Panel A shows raw detection rates by injected-error condition. Panel B shows fully adjusted odds ratios relative to the no-injection condition.

J.3 True Error Detection Efficiency

The increase in review time, however, does not necessarily imply that reviewers became less productive. We therefore examined true-error detection efficiency, defined for each review as the

proportion of true errors detected per minute of review time. We estimated the following ordinary least squares model:

$$E_{idt} = \beta_0 + \mathbf{C}'_{idt}\boldsymbol{\beta} + \theta O_{idt} + \gamma_t + \alpha_i + \delta_d + O_{idt}\mathbf{Z}'_i\boldsymbol{\eta} + \varepsilon_{idt}, \tag{6}$$

where E_{idt} denotes true-error detection efficiency for reviewer i evaluating document d during experimental period t; $\mathbf{C}_{idt}$ contains indicators for the low-, medium-, and high-injected-error conditions; and the omitted reference category is the control condition. O_{idt} denotes cumulative review order. The terms γ_t, α_i, and δ_d denote experimental-period, reviewer, and document fixed effects, respectively. The vector $\mathbf{Z}_i$ contains baseline familiarity, trust, and attention-related cognitive errors. Standard errors were clustered at the reviewer level. Reviews lasting more than 30 minutes were excluded as in the review time analysis above, and reviews with no true-error opportunities were excluded because detection could not be calculated. Figure 13 reports true-error detection efficiency across injected-error conditions.

The efficiency results show that reviewers in the low- and medium-injected-error conditions detected fewer true errors per unit time than reviewers in the control condition, whereas reviewers in the high-injected-error condition detected more true errors per unit time. In the raw data, efficiency decreased from 0.094 true errors detected per minute in the control condition to 0.083 and 0.078 in the low- and medium-injected-error conditions, respectively, but increased to 0.098 in the high-injected-error condition.

Efficiency did not differ significantly across injected-error conditions overall (omnibus $p = 0.326$). In the fully adjusted model, efficiency did not differ significantly between the high-injected-error and control conditions (adjusted mean difference, 0.007; 95% CI, −0.013 to 0.026; adjusted $p = 0.608$). Efficiency also did not differ significantly between the low-injected-error and control conditions (adjusted mean difference, −0.004; 95% CI, −0.015 to 0.008; adjusted $p = 0.608$) or between the medium-injected-error and control conditions (adjusted mean difference, −0.006; 95% CI, −0.029 to 0.017; adjusted $p = 0.608$).

Overall, these findings suggest that reviewers respond differently to different levels of error injection. High levels of injected errors lead reviewers to spend more time reviewing each article while also identifying more true errors per unit time. In contrast, low levels of injected errors reduce true-error detection without increasing review effort.

J.4 Injected-error Detection

Figure 14 reports reviewers' ability to identify injected errors. Detection rates were higher in the medium- and high-injected-error conditions than in the low-injected-error condition, although the differences were modest.

In the fully adjusted model, injected-error detection did not differ significantly across conditions overall (omnibus $p = 0.075$). Relative to the low-injected-error condition, reviewers in the medium-injected-error condition had higher odds of detecting an injected error (OR, 1.33; 95% CI, 0.98–1.80; adjusted $p = 0.102$). Reviewers in the high-injected-error condition also had higher odds of detecting an injected error (OR, 1.39; 95% CI, 1.04–1.86; unadjusted $p = 0.027$), although this difference was not statistically significant after adjustment for multiple comparisons (adjusted $p = 0.082$).

These findings provide suggestive evidence that reviewers may become more effective at identifying

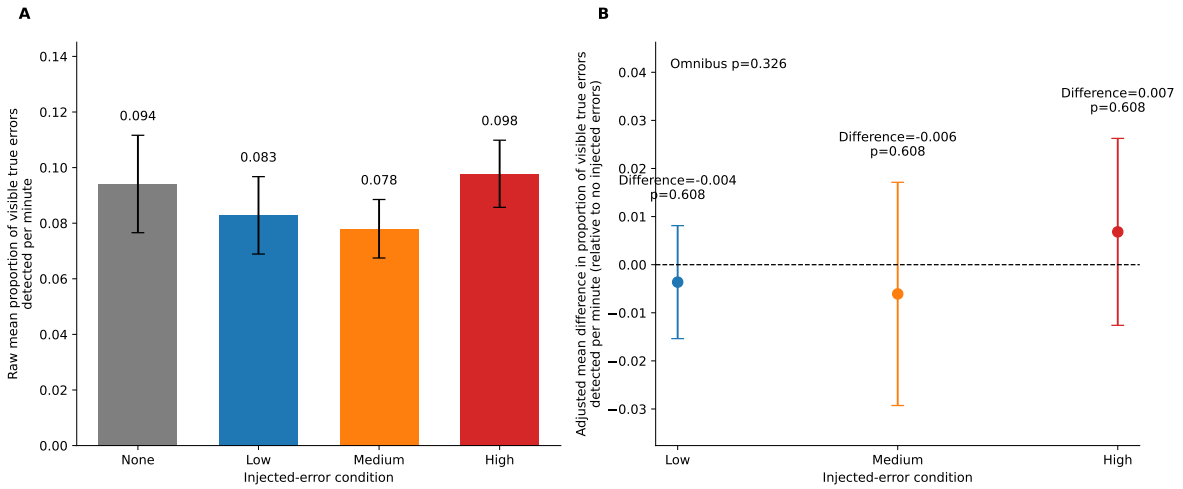

Figure 13: **True-error detection efficiency across injected-error frequencies.** Panel A reports raw efficiency. Panel B reports adjusted mean differences relative to the control condition. Error bars indicate 95% confidence intervals. p-values are corrected for multiple hypothesis testing.

injected errors when exposed to higher injected-error frequencies.

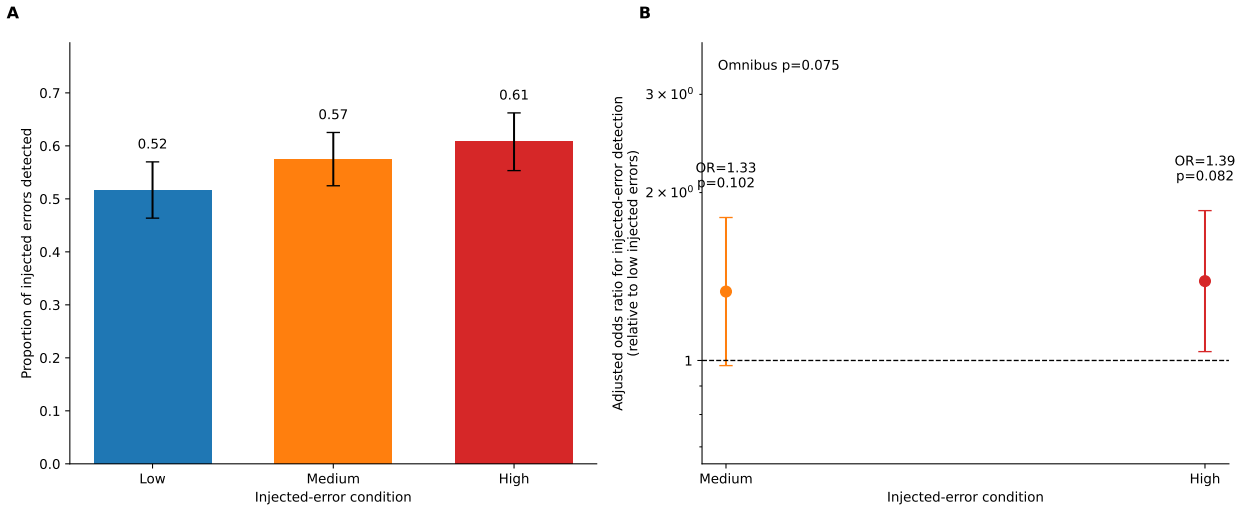

Figure 14: **Injected-error detection increases with injected-error frequency.** Panel A reports raw injected-error detection rates. Panel B reports adjusted odds ratios relative to the low-injected-error condition. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

J.5 Within-period Changes in Reviewer Behavior

This appendix reports the detailed exploratory analyses of within-period changes in reviewer behavior. For each reviewer and experimental period, we order completed reviews sequentially and divide them into quartiles. We compare reviews completed in the fourth quartile with those completed in the first quartile using the specification in Equation 5. The following subsections report the

outcome-specific estimates.

J.5.1 True-error Detection Within Review Period

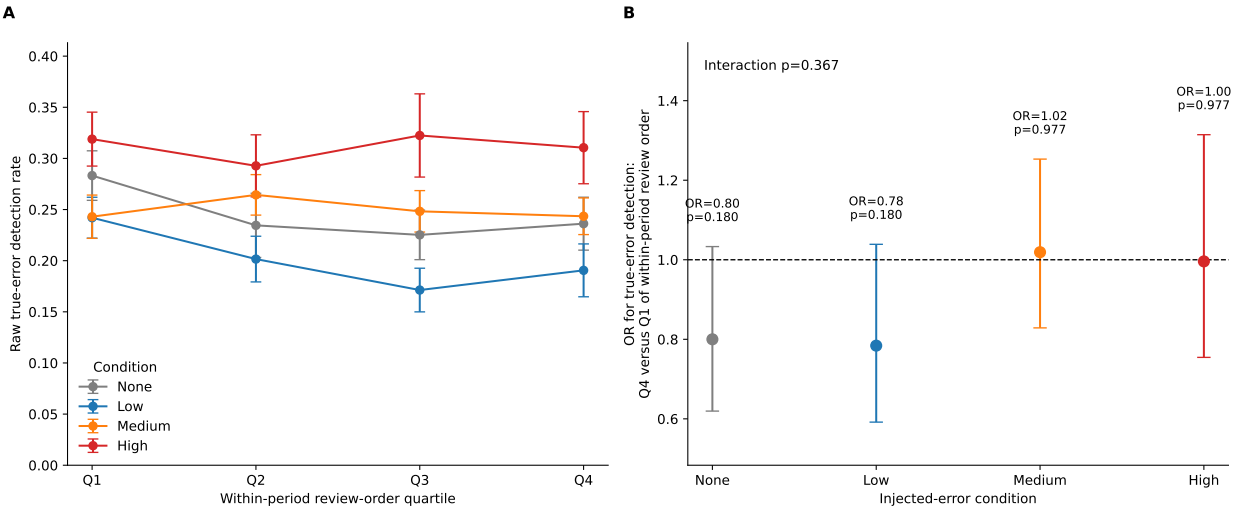

Figure 15: **Within-period trends in true-error detection.** Panel A reports raw true-error detection rates across quartiles of within-period review order. Panel B reports adjusted odds ratios comparing true-error detection in the fourth versus first quartile of within-period review order. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

Figure 15 reports within-period trends in true-error detection. The adjusted interaction between injected-error condition and the fourth-versus-first-quartile comparison was not statistically significant (interaction omnibus $p = 0.367$).

In the control condition, the odds of detecting a true error were lower in the fourth quartile than in the first quartile, but the difference was not statistically significant (OR, 0.80; 95% CI, 0.62–1.03; adjusted $p = 0.180$). The corresponding estimates were also not statistically significant in the low-injected-error condition (OR, 0.78; 95% CI, 0.59–1.04; adjusted $p = 0.180$), the medium-injected-error condition (OR, 1.02; 95% CI, 0.83–1.25; adjusted $p = 0.977$), or the high-injected-error condition (OR, 1.00; 95% CI, 0.75–1.31; adjusted $p = 0.977$). These estimates indicate that true-error detection did not change significantly over the course of a review period.

J.5.2 Review Time Within Review Period

Figure 16 reports within-period trends in review time. Reviewers generally completed articles more quickly in the fourth quartile than in the first quartile of a review period. In the control condition, review time was 24.3% lower in the fourth quartile than in the first quartile (95% CI, −34.3% to −12.7%; adjusted $p < 0.001$). Similar declines were observed in the low-injected-error condition (−24.2%; 95% CI, −35.3% to −11.0%; adjusted $p < 0.001$) and the high-injected-error condition (−23.9%; 95% CI, −32.3% to −14.6%; adjusted $p < 0.001$). Review time also declined in the medium-injected-error condition, although the estimate was smaller and not statistically significant (−10.9%; 95% CI, −23.9% to 4.3%; adjusted $p = 0.151$).

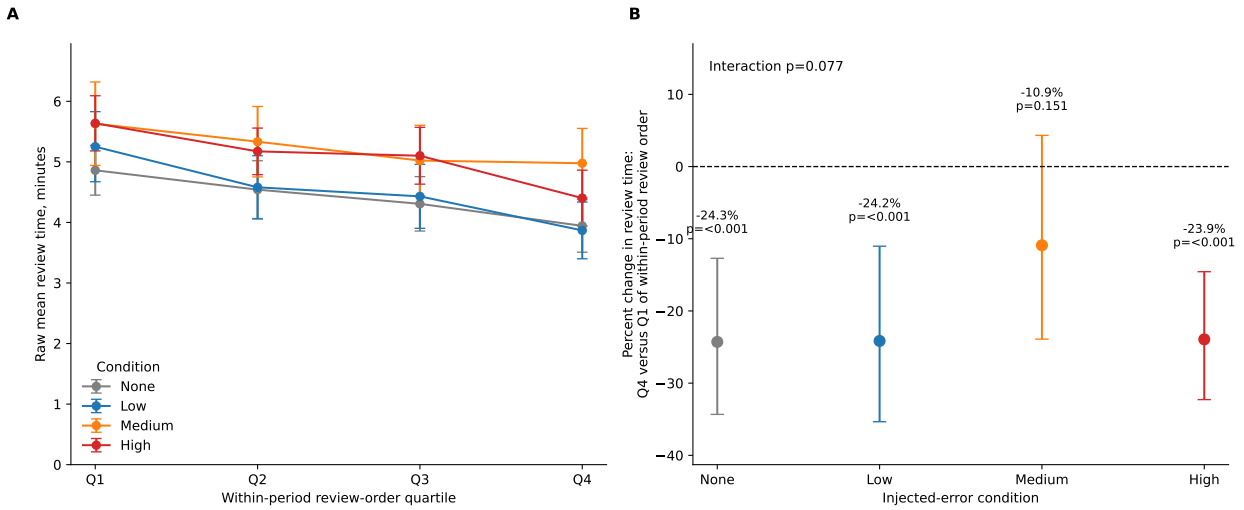

Figure 16: **Within-period trends in review time.** Panel A reports mean review times across quartiles of within-period review order. Panel B reports adjusted percentage changes in review time in the fourth versus first quartile of within-period review order. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

The interaction between injected-error condition and within-period review order was not statistically significant (interaction omnibus $p = 0.077$). Thus, review time declined within periods, but the magnitude of this decline was not statistically distinguishable across injected-error conditions.

J.5.3 Trust in AI Within Review Period

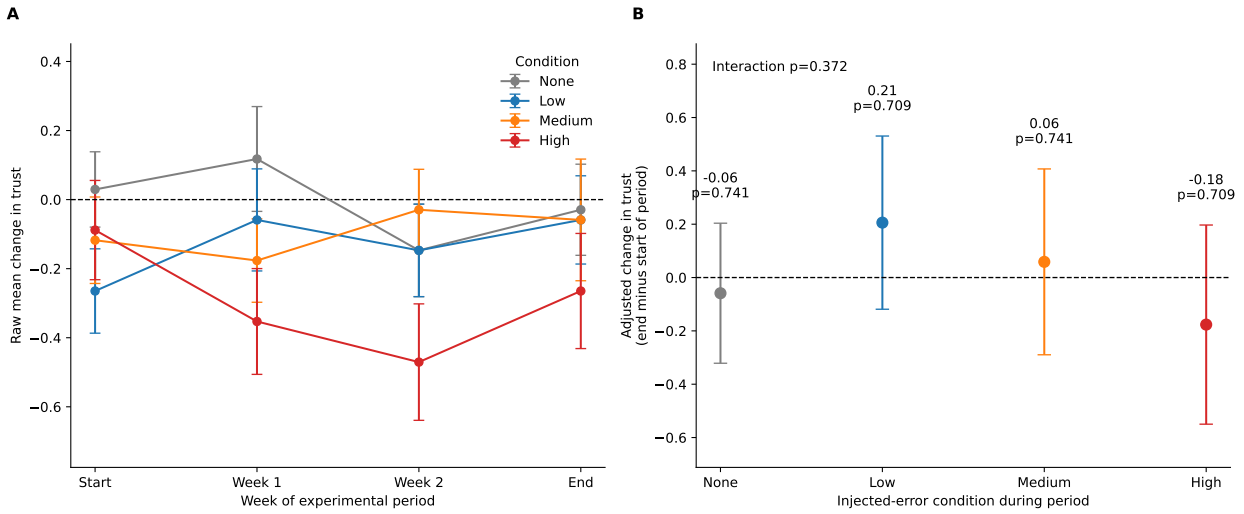

Figure 17: **Within-period trends in trust in AI.** Panel A reports raw mean change in trust across weeks of each experimental period. Panel B reports adjusted changes in trust from the start to the end of each experimental period. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

Figure 17 reports within-period trends in trust in AI. Trust did not change significantly from

the start to the end of a review period, and the interaction between injected-error condition and the end-versus-start comparison was not statistically significant (interaction omnibus $p = 0.37$).

In the control condition, trust was slightly lower at the end of the review period than at the start, but the difference was not statistically significant (adjusted change, −0.06; adjusted $p = 0.74$). Trust increased modestly in the low-injected-error condition (adjusted change, 0.21; adjusted $p = 0.71$) and the medium-injected-error condition (adjusted change, 0.06; adjusted $p = 0.74$), but neither estimate was statistically significant. In the high-injected-error condition, trust declined modestly from the start to the end of the period, but this change was also not statistically significant (adjusted change, −0.18; adjusted $p = 0.71$). These estimates indicate that trust in AI did not change significantly within review periods.

J.5.4 False-positive Flagging Within Review Period

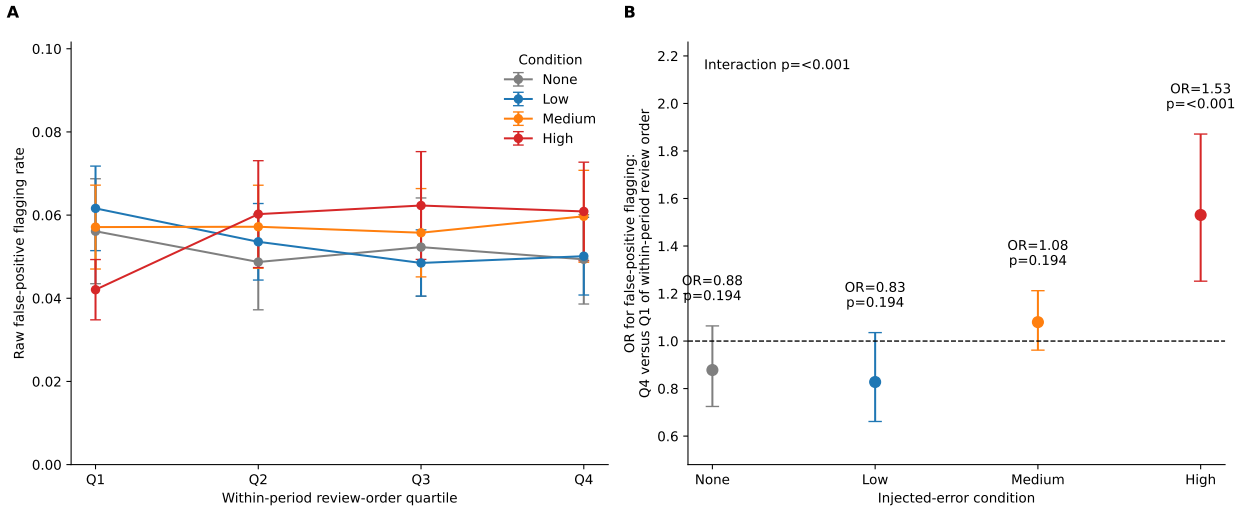

Figure 18: **Within-period trends in false-positive flagging.** Panel A reports raw false-positive flagging rates across quartiles of within-period review order. Panel B reports adjusted odds ratios comparing false-positive flagging in the fourth versus first quartile of within-period review order. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

Figure 18 reports within-period trends in false-positive flagging. The interaction between injected-error condition and the fourth-versus-first-quartile comparison was statistically significant (interaction omnibus $p < 0.001$).

In the control condition, the odds of falsely flagging an eligible correct cell were slightly lower in the fourth quartile than in the first quartile, but the difference was not statistically significant (OR, 0.88; 95% CI, 0.72–1.06; adjusted $p = 0.194$). A similar non-significant decline was observed in the low-injected-error condition (OR, 0.83; 95% CI, 0.66–1.04; adjusted $p = 0.194$). The medium-injected-error condition also showed no significant within-period change in false-positive flagging (OR, 1.08; 95% CI, 0.96–1.21; adjusted $p = 0.194$). By contrast, the high-injected-error condition was associated with significantly higher odds of false-positive flagging in the fourth quartile relative to the first quartile (OR, 1.53; 95% CI, 1.25–1.87; adjusted $p < 0.001$). Thus, false-positive flagging increased within the high-injected-error condition, while the other conditions did not exhibit

statistically significant within-period changes.

J.5.5 Association Between Injected-error and True-error Detection Within Review Period

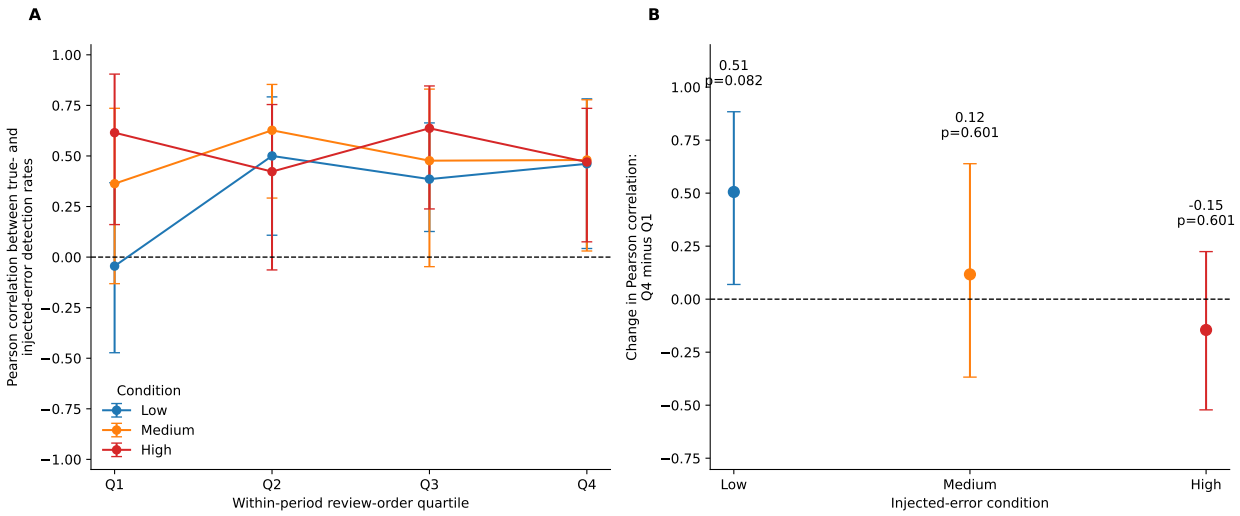

Figure 19: **Within-period trends in the association between true-error and injected-error detection.** Panel A reports the Pearson correlation between reviewer-level true-error detection rates and injected-error detection rates across quartiles of within-period review order. Panel B reports the change in this correlation from the first to the fourth quartile. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

Figure 19 examines whether injected-error detection remained informative about true-error detection as reviewers progressed through a review period. Because the control condition contained no injected-error opportunities, this analysis is restricted to the low-, medium-, and high-injected-error conditions.

The correlation between true-error detection and injected-error detection varied across quartiles, but there was no statistically significant change from the first to the fourth quartile in any injected-error condition. In the low-injected-error condition, the Pearson correlation increased from the first to the fourth quartile, but the change was not statistically significant (change in r, 0.51; adjusted $p = 0.082$). The medium-injected-error condition showed little change (change in r, 0.12; adjusted $p = 0.601$). The high-injected-error condition showed a modest decline in the correlation, but this change was also not statistically significant (change in r, -0.15; adjusted $p = 0.601$). These estimates indicate that the association between injected-error detection and true-error detection did not change significantly within periods.

J.5.6 Combined Error Detection Within Review Period

We also repeat the within-period analysis after pooling true-error opportunities and injected-error opportunities. This analysis measures whether detection of all known errors changed as reviewers progressed through a review period. Quartiles are defined within each reviewer-period block. The

descriptive panel reports all four quartiles, while the adjusted model compares the fourth quartile with the first quartile.

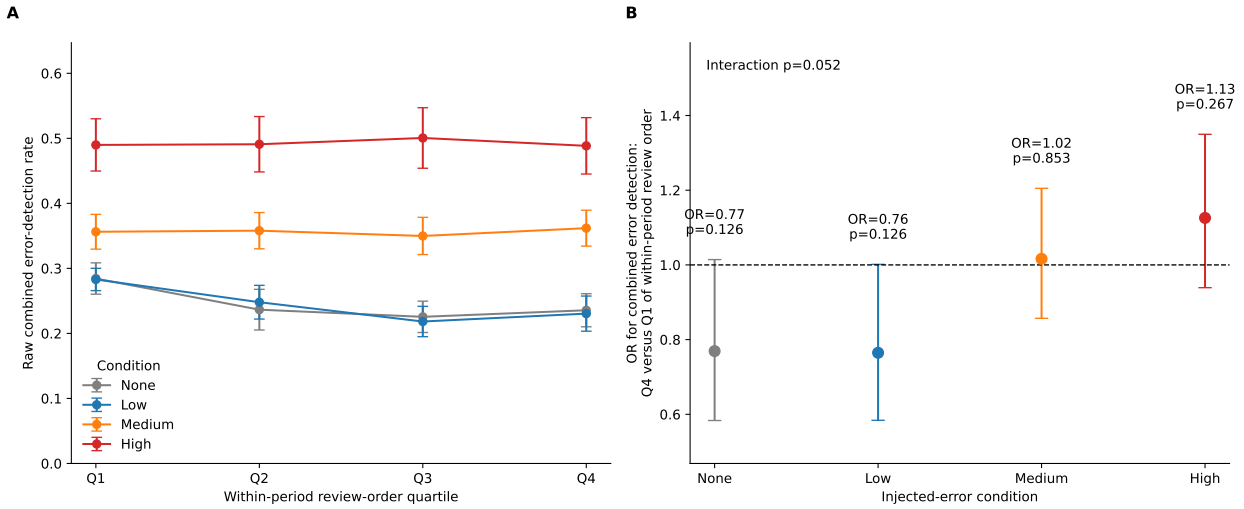

Figure 20: **Within-period trends in combined true-error and injected-error detection.** Panel A reports raw combined error-detection rates across quartiles of within-period review order. Panel B reports adjusted odds ratios comparing combined error detection in the fourth versus first quartile of within-period review order. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

Figure 20 reports within-period trends in combined error detection. The interaction between injected-error condition and the fourth-versus-first-quartile comparison was not statistically significant, although the estimate approached conventional significance (interaction omnibus $p = 0.052$).

In the control condition, the odds of detecting a known error were lower in the fourth quartile than in the first quartile, but the difference was not statistically significant (OR, 0.77; 95% CI, 0.58–1.01; adjusted $p = 0.126$). The low-injected-error condition showed a similar non-significant decline (OR, 0.76; 95% CI, 0.58–1.00; adjusted $p = 0.126$). Combined error detection was essentially unchanged in the medium-injected-error condition (OR, 1.02; 95% CI, 0.86–1.21; adjusted $p = 0.853$). In the high-injected-error condition, the odds of detecting a known error were higher in the fourth quartile than in the first quartile, but the estimate was not statistically significant after adjustment for multiple testing (OR, 1.13; 95% CI, 0.94–1.35; adjusted $p = 0.267$). These estimates indicate that combined error detection did not change significantly within periods.

J.5.7 Perceived Workload Within Review Period

We also examine whether perceived workload changed within review periods. Perceived workload was measured using the NASA Task Load Index composite. Because survey responses were collected weekly rather than at the article level, the analysis compares survey responses from the start and end of each three-week experimental period. The descriptive panel reports all available weekly measurements within each period.

Figure 21 reports within-period trends in perceived workload. Workload did not change significantly from the start to the end of a review period, and the interaction between injected-error

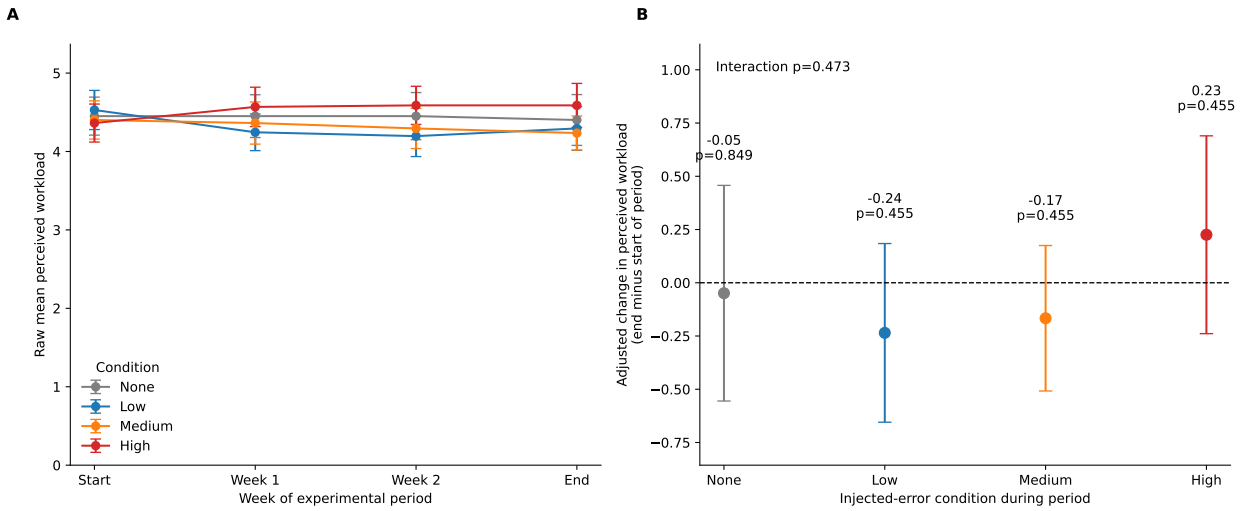

Figure 21: **Within-period trends in perceived workload.** Panel A reports raw mean perceived workload across weeks of each experimental period. Panel B reports adjusted changes in perceived workload from the start to the end of each experimental period. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

condition and the end-versus-start comparison was not statistically significant (interaction omnibus $p = 0.473$).

In the control condition, perceived workload was slightly lower at the end of the review period than at the start, but the difference was not statistically significant (adjusted change, −0.05; adjusted $p = 0.849$). Workload also declined modestly in the low-injected-error condition (adjusted change, −0.24; adjusted $p = 0.455$) and the medium-injected-error condition (adjusted change, −0.17; adjusted $p = 0.455$), but neither estimate was statistically significant. In the high-injected-error condition, perceived workload increased modestly from the start to the end of the period, but this change was also not statistically significant (adjusted change, 0.23; adjusted $p = 0.455$). These estimates indicate that perceived workload did not change significantly within review periods.

J.6 Heterogeneity Across Reviewers

We next examine whether reviewers with different baseline characteristics responded differently to the intervention. Figure 22 and Figure 23 report associations between baseline characteristics and perceived workload and trust, respectively.

Most moderator effects were small and statistically insignificant after correcting for multiple comparisons. For perceived workload, none of the condition-specific moderator coefficients remained statistically significant after Benjamini–Hochberg adjustment.

For trust in AI, the strongest association involved baseline everyday inattention. Among reviewers in the high-injected-error condition, greater baseline everyday inattention was associated with a more positive change in trust over time (adjusted coefficient, 0.17; 95% CI, 0.06 to 0.28; adjusted $p = 0.034$). Greater baseline adverse-event-report familiarity was also associated with a more positive change in trust in the high-injected-error condition before adjustment for multiple comparisons (adjusted coefficient, 0.11; 95% CI, 0.001 to 0.21; unadjusted $p = 0.049$), although

this association was not statistically significant after adjustment ($p = 0.292$). No other moderator remained statistically significant after correcting for multiple comparisons.

Taken together, these results provide limited evidence that reviewers with different baseline characteristics responded differently to the intervention.

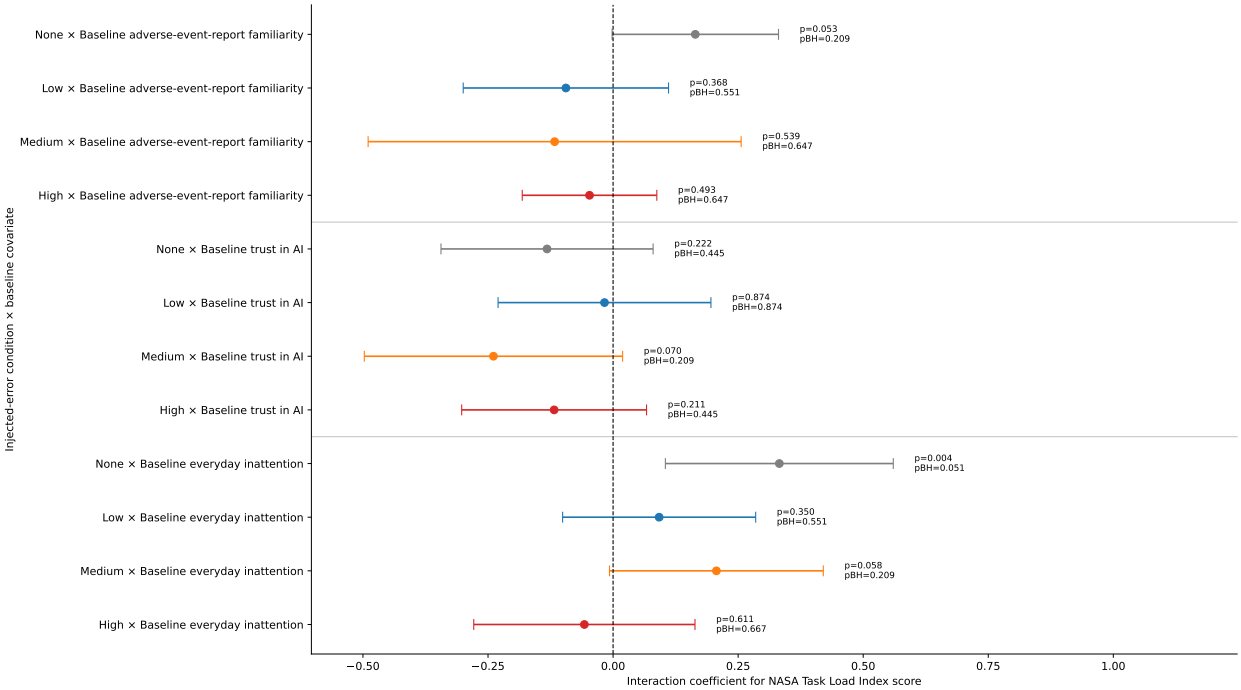

Figure 22: **Baseline characteristics are weakly associated with perceived workload.** Points report estimated moderator coefficients. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

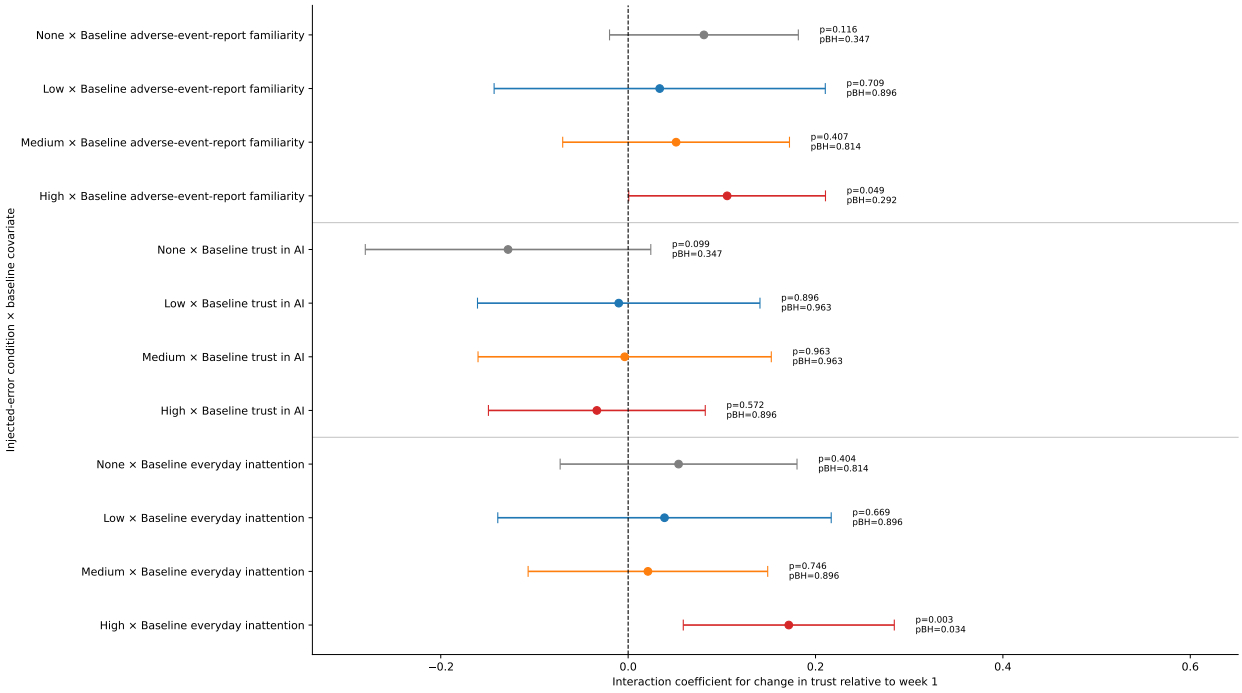

Figure 23: **Baseline characteristics are weakly associated with changes in trust.** Points report estimated moderator coefficients. Error bars indicate 95% confidence intervals. p-values are adjusted for multiple hypothesis testing.

J.7 Prior-period carryover of injected-error condition

In addition to testing for fatigue or learning effects within a review period, we examined across-period carryover by testing whether exposure to a condition in one period affected performance in subsequent periods. We tested whether the condition experienced in the immediately preceding period predicted current-period true-error detection, controlling for the current condition, study period, within-period review order, reviewer fixed effects, document fixed effects, and variable fixed effects.

Appendix Figure 24 shows the results. Across specifications, prior-period condition was not statistically associated with current-period true-error detection. In the fully adjusted model, the omnibus test for prior-period condition was not significant ($p = 0.130$). Raw current-period true-error detection rates were similar after prior no-injection, low-injection, and medium-injection periods, and somewhat lower after prior high-injection periods. Relative to prior no injection, prior low-frequency injection was associated with lower odds of current-period true-error detection, but the estimate was not statistically significant after multiple-comparison adjustment (OR, 0.79; 95% CI, 0.63–0.99; adjusted $p = 0.114$). Prior medium-frequency injection showed a similar non-significant pattern (OR, 0.82; 95% CI, 0.63–1.07; adjusted $p = 0.223$), while prior high-frequency injection showed no evidence of carryover (OR, 0.98; 95% CI, 0.84–1.16; adjusted $p = 0.848$). These results suggest that the main effects of error injection were primarily contemporaneous rather than driven by detectable carryover effects into later periods.

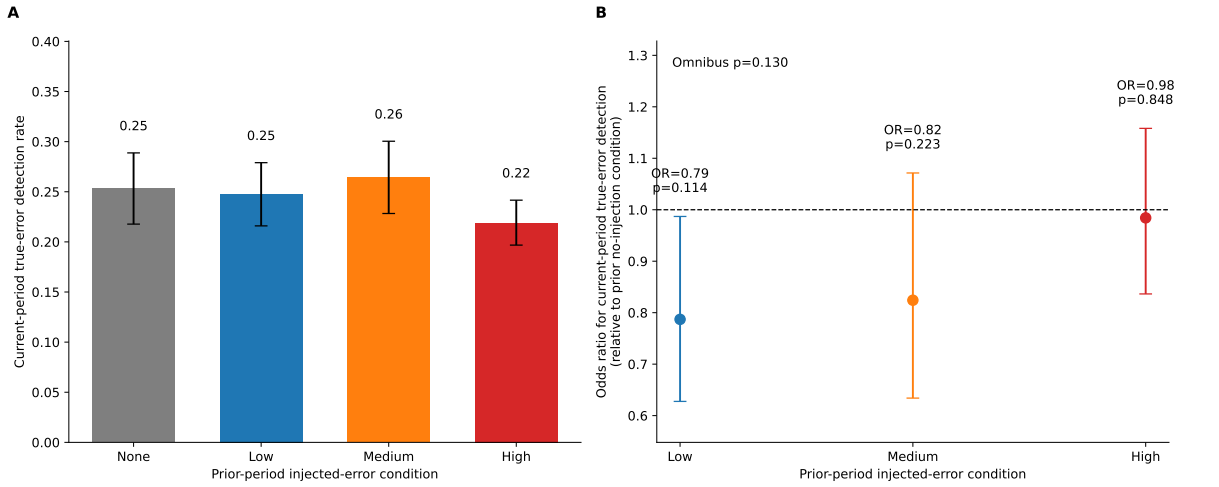

Figure 24: Prior-period carryover of injected-error condition on current-period true-error detection. Panel A shows raw current-period true-error detection rates by the condition experienced in the immediately preceding period. Panel B shows fully adjusted odds ratios for current-period true-error detection by prior-period condition, relative to prior no injection.