Adaptive auditing of AI systems with anytime-valid guarantees

Siyu Zhou[*,1] **Patrick Vossler**[*,1,3] **Venkatesh Sivaraman**[1,3] **Yifan Mai**[2] **Jean Feng**[1,3]
[1]University of California, San Francisco [2]Stanford University [3]Weill Cancer Hub West
[*]Equal contribution.

Abstract

A major bottleneck in characterizing the failure modes of generative AI systems is the cost and time of annotation and evaluation. Consequently, adaptive testing paradigms have gained popularity, where one opportunistically decides which cases and how many to annotate based on past results. While this framework is highly practical, its extreme flexibility makes it difficult to draw statistically rigorous conclusions, as it violates classical assumptions: the number of observations is typically limited (often 10 to 50 cases) and decisions regarding sampling and stopping are made in the midst of data collection rather than based a pre-specified rule. To characterize what statistical inferences can be drawn from highly adaptive audits, we introduce a hypothesis testing framework from two "dueling" perspectives: (i) the model's null that asserts there is no failure mode with performance below a target threshold versus (ii) the auditor's null that asserts they have a sampling strategy that will uncover a failure mode. Leveraging Safe Anytime-Valid Inference (SAVI), we formalize the auditor as conducting "testing by betting", which translates into simultaneous e-processes for testing the dueling null hypotheses. Furthermore, *if* the auditor is sufficiently powerful, we prove that these two hypotheses are asymptotically inverses of each other, in that passage of such an audit certifies the AI system as being globally robust. Empirically, we demonstrate that our proposed testing procedures maintain anytime-valid Type-I error control, outperform pre-specified testing methods, and can reach statistically rigorous conclusions sometimes with as few as 20 observations.

1 Introduction

AI systems are known to exhibit various generalization failures across subgroups and input types, often described as the "jagged frontier." Because gold-standard evaluations for AI (agent) outputs can often be time- and/or cost-consuming [Bandel et al., 2026], a common practice is to construct small, adaptively selected test suites to conduct more targeted probing of potential failure modes [Husain and Shankar, 2026, Anthropic, 2026, Yan, 2025]. The practical appeal of adaptive testing is that one can decide in real-time which samples to annotate and how many based on results from past annotations; approaches include checklist-style behavioral tests that probe specific linguistic capabilities [Ribeiro et al., 2020]; human-in-the-loop systems where the practitioner interactively refines test cases alongside model-generated candidates [Ribeiro and Lundberg, 2022, Gao et al., 2023]; visualization interfaces to audit algorithmic fairness [Cabrera et al., 2019]; and adversarial generation, where practitioners try to elicit worst-case behaviors via red-teaming [Perez et al., 2022, Ganguli et al., 2022, Lee et al., 2023, Li et al., 2024]. This practice even extends beyond AI, such as stress-testing devices across worst-case scenarios [U.S. Food and Drug Administration, 2020].

Despite its widespread use, the flexibility of adaptive testing workflows raises a fundamental statistical question: can a small number of adaptively chosen observations support rigorous inference about

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

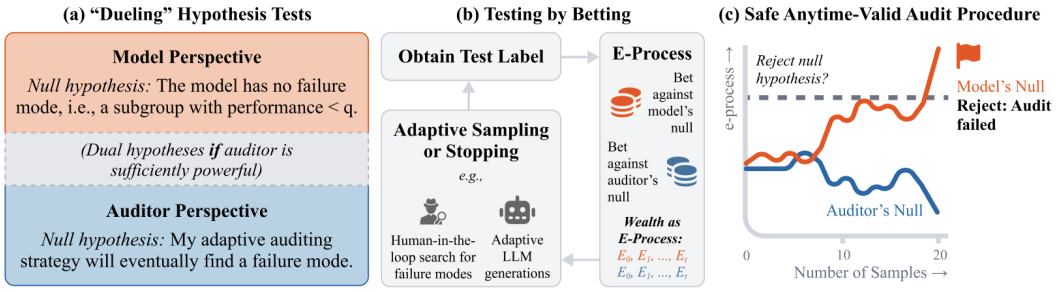


Figure 1: We (a) formalize flexible audits of failure modes in AI systems as dueling hypothesis tests, (b) introduce a "testing-by-betting" framework that is suitable for settings where sampling and stopping strategies are decided in the midst of data collection and not pre-defined, and (c) represent adaptive bets as dual e-processes with safe anytime-valid guarantees.

system robustness? Standard statistical inference typically assumes both a pre-specified sampling scheme and sample sizes that are large enough for asymptotic theory to apply. Adaptive testing violates both: AI practitioners often curate as few as 10 to 50 test cases [Husain and Shankar, 2026, Yan, 2025], with the sampling and stopping strategy decided in the midst of data collection rather than based on a pre-specified rule. Such testing regimes are traditionally disallowed as they can substantially inflate Type I error; for instance, if one continuously peeks at the p-value, the test is guaranteed to reject with probability one under the null [Ramdas and Wang, 2025]. Even more fundamentally, it is difficult to formulate what exactly is being tested; for instance, arbitrary sampling practices cannot be used to test a model's average performance. *Thus, the goal of this work is to (i) understand what hypothesis tests can be assessed under highly flexible testing regimes and (ii) construct corresponding testing procedures that are universally valid with finite-sample guarantees.*

Our proposal begins with formalizing AI robustness auditing as conducting two "dueling" hypothesis tests, one from the perspective of the "model" and one from the perspective of the "auditor" (Figure 1(a)). The model's null hypothesis makes the omnibus assertion that no failure modes exist, in that every sufficiently large subgroup meets a minimum performance threshold; rejection of the model's null means that the AI model is not robust. The auditor's null hypothesis asserts that they have a strategy that will eventually identify a failure mode; rejection of the auditor's null implies passage of this specific audit. While these two null hypotheses were specifically constructed to be testable in practice, they are generally not exact complements. Thus, they cannot conduct the ideal audit that yields a binary conclusion of whether an AI system is or is not robust. But *if* one has a powerful auditor that is guaranteed to find a failure mode if one truly exists, we prove that, asymptotically, the two null hypotheses are complements and thus correspond to the ideal audit. Through this formalization, we clarify what can be tested in practice, what assumptions are additionally needed to conduct the ideal audit in theory, and how one should interpret and conduct audits.

We then leverage Safe Anytime-Valid Inference (SAVI) methods to frame the auditor as conducting a "Testing by Betting" procedure, in which the auditor is allowed to make adaptive bets against any sequence of subgroups that the auditor suspects could be failure modes and where under the null, there is no winning betting strategy (Figure 1 b, c) [Grünwald et al., 2024, Ramdas and Wang, 2025]. We investigate various mathematical translations of these bets into e-processes—the SAVI analog of the p-value in the sequential setting—which can be interpreted as the accumulation of wealth from the auditor's bets and, crucially, provide valid finite-sample Type-I error control under arbitrary adaptive sampling and optional stopping. We begin with the simplest option of the likelihood ratio, then define more adaptive formulations based on universal inference, and finally introduce changepoint-based formulations.

We make the following contributions:

1. We formalize AI auditing as conducting dueling hypothesis tests between the model's null and the auditor's null. We prove these are asymptotically complements under sufficiently powerful auditing strategies.
2. We introduce dual e-processes that provide stopping bounds for simultaneous failure mode detection and audit certification, with SAVI guarantees.
3. We provide empirical results demonstrating how adaptive testing paired with the e-process procedures can indeed provide Type-I error control and make rigorous conclusions much faster than pre-specified auditing strategies, sometimes with as few as 20 observations.

2 Related Work

LLM Evaluation and Adaptive Testing. Many free-form LLM outputs are expensive and time-consuming to evaluate, e.g., multiple subagent calls or need manual annotation by experts [Ghosh et al., 2026]. Consequently, beyond large-scale benchmarking efforts of LLMs [Liang et al., 2023], a growing body of research focuses on uncovering failure modes of LLMs through highly targeted adaptive or active testing, in which results obtained in previous rounds inform which observations to label next [Ribeiro and Lundberg, 2022, Gao et al., 2023, Li et al., 2024, Vivek et al., 2024]. These adaptive testing methods often involve human-in-the-loop, sometimes assisted by automatic systems; in the most extreme scenario, active testing is akin to red-teaming, where one constructs worst-case test scenarios [Ganguli et al., 2022]. Due to their resource/cost-efficient nature, adaptive testing methods are also commonly used by AI practitioners to evaluate AI pipelines in target areas, where common recommendations are to assemble diverse test suites with 10–50 cases to cover edge cases and failure modes [Husain and Shankar, 2026, Anthropic, 2026]. However, there is currently no statistical framework that formalizes what exactly is and can be the target of (statistical) inference when one conducts highly adaptive and targeted testing with unknown sampling schemes, as this violates assumptions in classical active testing settings that require a pre-specified sampling scheme [Kuang et al., 2025]. Moreover, existing works [Yan and Zhang, 2022, Berrada et al., 2025] often focus on active estimation of the average loss in the population of interest or in pre-specified groups instead of an omnibus testing of the existence of failure modes. This work provides such a testing framework by allowing for arbitrary sampling and stopping schemes.

E-values and E-processes. Because classical p-value and hypothesis testing procedures place rigid constraints and assumptions on the data generating mechanism, recent works have proposed an alternative approach based on e-values. E-values, whose properties are derived from (conditional) expectations rather than probability distributions, have the advantage of providing anytime-valid testing with finite-sample Type I error control under any unknown sampling schemes [Grünwald et al., 2024, Ramdas and Wang, 2025]. Furthermore, e-values have been extended to the online setting under optional stopping, with e-processes for sequential testing and e-detectors for sequential changepoint detection [Shin et al., 2024, Zhou et al., 2026]. These frameworks can be viewed as testing by betting, in which the tester places a bet at each iteration and, under the null, the expected return is (no greater than) zero [Shafer, 2021, Chugg et al., 2023]. While there is growing interest in using e-values to improve generated outputs from AI systems [Sadhuka et al., 2025], there are currently no e-value frameworks for highly targeted and adaptive audits of AI systems, as the exact hypothesis tests in these settings have yet to be formalized.

Subgroup and Fairness Testing. The problem of auditing failure modes of AI systems falls under the broader research topic of testing and identifying subgroups with large performance disparities [Chung et al., 2019, Eyuboglu et al., 2023, Feng et al., 2023, Singh et al., 2023, Subbaswamy et al., 2024, Zeng et al., 2026] and algorithmic fairness [Chouldechova and Roth, 2020, Mitchell et al., 2021]. Nevertheless, the existing literature either studies the offline (batch) settings or active labeling with pre-specified sampling or subgroups [Hartmann et al., 2026]. Our framework differs in its generality by allowing for statistically rigorous inference under active testing where neither the subgroups nor sampling strategy are pre-specified.

3 Methods

To formalize adaptive audits of AI system robustness, the following section addresses the two main technical challenges: (i) what null hypotheses can be feasibly tested and (ii) what testing procedures provide safe anytime-valid inference? Due to space constraints, we refer the reader to the Appendix for implementation details and detailed proofs.

Notation. Let $(\mathcal{X}, \mathcal{F}, \mu)$ denote a probability space with input space $\mathcal{X}$, σ-algebra $\mathcal{F}$, and probability measure μ. For query $X \in \mathcal{X}$, let random variable Y denote the score for the AI system's response (e.g., correctness), which we assume to be IID conditional on X. For a measurable subset $S \in \mathcal{F}$ with prevalence $\mu(S) > 0$, the AI system's average score for subgroup S is denoted $V(S) := \mathbb{E}[Y \mid X \in S]$ (we presume higher scores are better). For fixed $\epsilon \in (0, 1]$, let $\mathcal{S}_\epsilon := \{S \in \mathcal{F} : \mu(S) \geq \epsilon\}$ denote all subgroups with mass at least ϵ. A *failure mode* is a subgroup $S \in \mathcal{S}_\epsilon$ on which $V(S)$ falls below a target threshold q. For any positive integer r, $[r]$ corresponds to the sequence $\{1, 2, \ldots, r\}$.

3.1 Dueling hypothesis tests for auditing AI robustness

Defining a suitable hypothesis testing framework requires overcoming a fundamental asymmetry in adaptive model auditing: while identifying a failure mode proves a lack of robustness, passing a targeted test suite does not certify its robustness. This is true from both the software engineering perspective of unit tests as well as the Fisherian perspective of statistical hypothesis testing [Christensen, 2005]. Furthermore, because adaptive sampling opportunistically seeks out difficult cases, it cannot make conclusions about typical estimands like average or subgroup-specific performance.

We resolve this by framing AI robustness audits as conducting two "dueling" hypothesis tests: the *model's null* makes the omnibus assertion that no failure modes exist, while the *auditor's null* asserts that their specific auditing strategy will eventually find a failure mode. We then investigate conditions under which testing these two null hypotheses corresponds to the ideal audit, where passage of the audit occurs if and only if the AI system is globally robust.

3.1.1 The Model's hypothesis

The model's null hypothesis states that it performs well across all subgroups with prevalence at least ϵ, while the alternative states that there exists one or more failure modes. This can be mathematically formalized as

$$\begin{aligned} H_0^{\texttt{mod}}: &\quad \forall S \in \mathcal{S}_\epsilon, \quad V(S) \geq q, \\ H_1^{\texttt{mod}}: &\quad \exists S \in \mathcal{S}_\epsilon \text{ such that } V(S) < q. \end{aligned} \tag{1}$$

We emphasize that this is a subgroup-uniform testing problem rather than an estimation problem: the procedure does not need to recover the worst-case subgroup nor its accuracy.

The closest analog to this test appears in the *multicalibration* literature [Hebert-Johnson et al., 2018, Feng et al., 2023], which studies whether a model is well-calibrated over a collection of subgroups. Similar questions are also considered in the Distributionally Robust Optimization (DRO) literature where one may be interested in the worst-case subgroup performance [Subbaswamy et al., 2021]. The key difference in this work is that we study adaptive anytime testing, whereas prior works focused on the batch setting with pre-specified sampling schemes.

3.1.2 The Auditor's hypothesis

The auditor's null hypothesis asserts that they have an unspecified, adaptive, and potentially adversarial auditing strategy that will eventually uncover a failure mode. Formally, auditing strategy π is defined as a function of the natural filtration at time $t-1$, denoted $\mathcal{F}_{t-1}$, to the adaptively selected subgroup at time t, denoted $S_t^\pi \in \mathcal{S}_\epsilon$. At each time t, the auditor bets against a subgroup S_t^π, samples an observation X_t from it (with respect to μ restricted to S_t^π), and then obtains the corresponding score Y_t. Formally, we represent the auditor's hypothesis test as

$$\begin{aligned} H_0^{\texttt{aud},[m]}: &\exists \tau \in [m],\ s.t.\ \forall t \geq \tau,\ V(S_t^\pi) < q, \\ H_1^{\texttt{aud},[m]}: &\forall \tau \in [m],\ \exists t \geq \tau,\ s.t.\ V(S_t^\pi) \geq q. \end{aligned} \tag{2}$$

Note that (2) is defined with respect to a hyperparameter m, which states that the auditor has a budget of size m to find the failure mode. This budget is necessary to ensure the null hypothesis is testable in practice (otherwise one would have to wait forever). Nevertheless, neither data collection nor the auditor's adaptive sampling strategy need to be frozen at time m, nor should it be. Instead, as discussed later, budget m is simply the time at which we may begin seriously testing the auditor's null hypothesis.

3.1.3 Duality of the model's and auditor's null

In practice, one can only test the model's and auditor's null hypotheses. Nevertheless, because the auditor's null only makes a statement about a specific audit strategy, it is natural to ask what conditions are needed to certify that an AI system is *globally* robust. We show that this is possible if the auditor's strategy is *asymptotically consistent*, as defined below.

Definition 1 (Asymptotic consistency)**.** *The auditor's strategy π is asymptotically consistent if the performance of the selected subgroups converges to below q, i.e., $\lim_{t\to\infty} V(S_t^\pi) < q$, whenever there exists a subgroup $S^* \in \mathcal{S}_\epsilon$ with performance below q.*

Many auditing strategies are asymptotically consistent. The simplest option is to conduct IID sampling and estimate $V(x) = \mathbb{E}(Y|X = x)$ with some model $\hat{V}_n$. Standard results regarding the convergence of such estimators imply this asymptotic consistency condition [Anthony and Bartlett, 1999, Biau and Cadre, 2021, Scornet et al., 2015]. A more efficient option is to have the auditor use active learning methods, which sample observations to maximize learning and provide similar consistency results (see e.g., Goetz et al. [2018] and Zhu and Nowak [2022]). In fact, the bandit literature may offer solutions that best match the problem faced in adaptive auditing: the goal of finding the worst performing subgroup is equivalent to the continuous multivariate bandit with arms corresponding to X and the reward corresponding to the negative score $-Y$ [Auer et al., 2002, Srinivas et al., 2009, Zhou et al., 2020]. Regret bounds from such algorithms imply the asymptotic consistency condition, which inspires the use of an Upper Confidence Bound (UCB)-based auditor in Section 4.

Conditional on having a consistent auditing strategy, we have the following duality result that states that the model's and auditor's null hypotheses are asymptotically complements.

Theorem 1. *Conditional on an asymptotically consistent auditing strategy π, the model's and auditor's hypotheses are complementary asymptotically as the budget m increases, i.e.,*

$$\lim_{m\to\infty} H_1^{\texttt{aud},[m]} \text{ true} \iff H_0^{\texttt{mod}} \text{ true} \quad \text{and} \quad \lim_{m\to\infty} H_0^{\texttt{aud},[m]} \text{ true} \iff H_1^{\texttt{mod}} \text{ true}.$$

Theorem 1 implies that if the auditor is strong enough, testing the two "dueling" hypotheses with a sufficiently large budget m can in fact be interpreted as providing *dual* stopping bounds for certifying or rejecting an AI system as being globally robust. Such dual stopping bounds can also be viewed as a generalization of those used in clinical trials to declare efficacy versus futility of a treatment [Pocock, 1977, O'Brien and Fleming, 1979]. The key difference is that we focus on an omnibus test with respect to all subgroups $\mathcal{S}_\epsilon$, whereas classical dual stopping bounds are solely concerned with the average treatment effect with respect to the full population.

Practically speaking, this result suggests leveraging both expert- and data-driven methods when auditing AI systems, as the former can provide useful prior knowledge in settings with limited data while the latter can ensure asymptotic consistency. At the same time, this result warns against making global robustness claims when the auditor is weak, echoing recent concerns that poorly designed red-teaming exercises may be akin to "security theater" [Feffer et al., 2024].

3.2 Dual safe anytime-valid testing procedures

We now investigate procedures that translate fully adaptive sampling schemes and optional stopping into statistics for testing the model's and auditor's hypotheses while providing finite-sample guarantees. Motivated by the "Testing by Betting" framework [Shafer, 2021], we formulate testing procedures as e-processes [Ramdas and Wang, 2025], a key mathematical object from the SAVI literature:

Definition 2 (E-process)**.** *A sequence $(E_t)_{t\geq 0}$ adapted to a filtration $(\mathcal{F}_t)$ is an* e-process *with respect to H_0 if $\mathbb{E}_P[E_\tau] \leq 1$ for any stopping time τ and any $P \in H_0$. Equivalently, $(E_t)_{t\geq 0}$ is an e-process if there exists a test supermartingale family $(M_t^P)_{P\in H_0}$ such that, $\forall P \in H_0$, (i) $M_t^P \geq 0$ P-almost surely, (ii) M_t^P is a supermartingale under P, and (iii) $\mathbb{E}_P[M_0^P] \leq 1$, and that $E_t \leq M_t^P$ P-almost surely.*

From a testing perspective, the key property of e-processes is that they are guaranteed to control the Type I error rate under adaptive sampling and optional stopping by Ville's inequality. That is, for any $P \in H_0$ and $\mathcal{F}$-stopping rule τ, an e-process stopped at threshold $1/\alpha$ has a Type I error rate no more than α, i.e.,

$$P\left(E_\tau \geq \frac{1}{\alpha}\right) \leq \alpha. \tag{3}$$

As such, we next present various e-processes for testing the model's and auditor's hypotheses, beginning with the simplest option of the likelihood ratio and building up to more complex constructions. For ease of exposition, this section focuses on binary scores Y (e.g., accuracy) but the results extend to continuous or multiclass scoring.

3.2.1 Testing the model's null

To test the model's null hypothesis, we consider the likelihood ratio process, an adaptive version based on Universal Inference (UI) [Wasserman et al., 2020], and extensions to both procedures by additionally integrating changepoint detection [Shiryaev, 1963, Roberts, 1966, Shin et al., 2024].

Option 1: Likelihood ratio process (`LR`). One of the most well-known approaches to construct an e-process is to use likelihood ratios, which is based on the classical Sequential Probability Ratio Test (SPRT) [Wald, 1945]. In particular, let $p(Y;\gamma)$ be the likelihood of a Bernoulli distribution with probability γ, and define the `LR` test statistic as $E_t^{\text{LR},\texttt{mod}} := E_{1:t}^{\text{LR},\texttt{mod}}$ where

$$E_{j:t}^{\text{LR},\texttt{mod}} = \prod_{k=j}^{t} \frac{p(Y_k; q-\delta)}{p(Y_k; q)}. \tag{4}$$

Here, $\delta > 0$ is a fixed user-specified separation parameter. When the auditor samples subgroups with $V(S_t^\pi) \leq q - \delta$, the process has positive expected log-growth against the boundary value q. Subgroups with $V(S_t^\pi) \in (q-\delta, q)$ remain failures under $H_1^{\texttt{mod}}$ but may produce slow growth, if any. Under the null $H_0^{\texttt{mod}}$, the test statistic $E_t^{\text{LR},\texttt{mod}}$ is an e-process, with no expected growth. Therefore rejecting $H_0^{\texttt{mod}}$ at time t if $E_t^{\text{LR},\texttt{mod}} \geq \frac{1}{\alpha}$ is an anytime-valid level-α sequential test.

Option 2: Likelihood Ratios with UI (`LR-UI`). While simple to construct, the disadvantage of the `LR` procedure above is that the fixed probability $q - \delta$ in (4) can be suboptimal for detecting a particular alternative in $H_1^{\texttt{mod}}$. To improve its power, we introduce an adaptive version denoted `LR-UI` that draws on UI to define the test statistic $E_t^{\text{LR-UI},\texttt{mod}} := E_{1:t}^{\text{LR-UI},\texttt{mod}}$, where

$$E_{j:t}^{\text{LR-UI},\texttt{mod}} = \prod_{k=j}^{t} \frac{p\left(Y_k; \hat{\gamma}_{k-1}^{\texttt{mod}}\right)}{p\left(Y_k; q\right)}. \tag{5}$$

Here $\hat{\gamma}_{t-1}^{\texttt{mod}}$ is any estimator of $V(X_t)$ with respect to the natural filtration $\mathcal{F}_{t-1}$ that outputs a probability consistent with $H_1^{\texttt{mod}}$. By using an estimator, one can view the auditor as placing an adaptive bet on not only which subgroup is a potential failure mode, but also its likely failure rate, in the hopes to further optimize its gain in the testing-by-betting framework. Because Y_t and $\hat{\gamma}_{t-1}^{\texttt{mod}}$ are conditionally independent given $\mathcal{F}_{t-1}$, $E_t^{\text{LR-UI},\texttt{mod}}$ is a test supermartingale and thus an e-process for $H_0^{\texttt{mod}}$.

Options 3 and 4: SR Likelihood ratio processes (`SR-LR`, `SR-LR-UI`). `LR` and `LR-UI` are optimal when the auditor identifies failure modes from the very beginning, but an auditor may require multiple attempts (e.g., a learning period) in practice. Consequently, we consider integrating changepoint detection techniques into the aforementioned e-processes, so that the testing procedures are powered to detect the emergence of failure modes in the adaptive testing sequence.

In particular, we will consider an extension based on the Shiryaev-Roberts (SR) procedure [Shiryaev, 1963, Roberts, 1966], which is classically defined as the sum of LR processes starting from all possible candidate changepoints and has been more recently studied under the umbrella of e-detectors [Shin et al., 2024]. Because the weighted sum of e-processes is also an e-process, we define the test statistics `SR-LR` and `SR-LR-UI` as the e-processes

$$E_t^{\text{SR-LR},\texttt{mod}} = \sum_{j=1}^{t} w_j E_{j:t}^{\text{LR},\texttt{mod}}, \quad E_t^{\text{SR-LR-UI},\texttt{mod}} = \sum_{j=1}^{t} w_j E_{j:t}^{\text{LR-UI},\texttt{mod}}.$$

for pre-defined weights $\{w_j : j = 1, 2, \cdots\}$ that sum to one. In the betting framework, the SR extension equates to the auditor distributing their wagers across all candidate changepoints to hedge against the unknown delay before a failure mode emerges.

3.2.2 Testing the auditor's null

To test the auditor's null hypothesis, we cannot directly apply the aforementioned solutions because the auditor's null has a complex composite structure whereas the model's null is much simpler. The challenge is most apparent when viewing the auditor's null as a changepoint statement. In this formulation, the null asserts the presence of a drop in the subgroup scores - a changepoint - *within* the

allotted budget m, whereas the alternative asserts the absence of such a drop. Rejecting the null prior to observing *all* m observations is therefore risky, since doing so entails an unwarranted extrapolation about the future of an auditor adaptive to the *full* past trajectory.

As such, one may ask if it is even possible to meaningfully test the auditor's null prior to time m. We prove that this is indeed impossible, in that there is no anytime-valid testing procedure that can achieve higher power than its Type I error prior to the m-th observation.

Theorem 2. *Let $\phi(t)$ be any anytime-valid testing procedure for the auditor's null hypotheses such that the Type I error rate is controlled at α, i.e., $\sup_{m' \in [m], t < m'} \mathbb{E}\left(\phi(t) | H_0^{\texttt{aud},m'} \text{ is true}\right) \leq \alpha$ where $H_0^{\texttt{aud},m'}$ is the singleton analog of the auditor's null hypothesis. Then, the power prior to time m is also no more than α, i.e., $\sup_{m' \in [m], t < m'} \mathbb{E}\left(\phi(t) | H_1^{\texttt{aud},m'} \text{ is true}\right) \leq \alpha$.*

Given this result, we can simplify the testing procedure to wait until the end of the budget window m. Then starting at time m, we can test the simplified null hypothesis

$$\begin{aligned} H_0^{\texttt{aud},m} &: \forall t \geq m,\ V(S_t^\pi) < q, \\ H_1^{\texttt{aud},m} &: \exists t \geq m,\ s.t.\ V(S_t^\pi) \geq q, \end{aligned} \tag{6}$$

for which the testing procedures from the previous section are now applicable.

Option 1. Likelihood Ratio Process (`LR`). The first option is to define a likelihood ratio process similar to (4), but starting from time m and with the larger probability parameter in the numerator. The `LR` test statistic for the auditor's null is thus

$$E_t^{\text{LR},\texttt{aud}} = \begin{cases} \prod_{k=m}^{t} \frac{p(Y_k; q+\delta')}{p(Y_k; q)}, & \text{if } t \geq m, \\ 1, & \text{otherwise,} \end{cases} \tag{7}$$

for some fixed tolerance $\delta' > 0$.

Option 2. Likelihood Ratios with UI (`LR-UI`). Alternatively, we can again use an adaptive version of the likelihood ratio process to avoid specifying a fixed alternative. That is, the `LR-UI` test statistic for the auditor's null is

$$E_t^{\text{LR-UI},\texttt{aud}} = \begin{cases} \prod_{k=m}^{t} \frac{p\left(Y_k; \hat{\gamma}_{k-1}^{\texttt{aud}}\right)}{p(Y_k; q)}, & \text{if } t \geq m, \\ 1, & \text{otherwise,} \end{cases} \tag{8}$$

where $\hat{\gamma}_{t-1}^{\texttt{aud}}$ is an estimator of $V(X_t)$ with respect to the natural filtration $\mathcal{F}_{t-1}$ under $H_1^{\texttt{aud},m}$.

3.2.3 Dual testing

Combining the e-processes for the model's and auditor's nulls yields dual safe anytime-valid procedures for any adaptive auditor (Figure 1; Algorithm 1). At each round t, the model's test statistic is updated; once $t \geq m$, the auditor's test statistic is monitored alongside it. The procedure stops as soon as either statistic exceeds $1/\alpha$. Because at most one of the two nulls is true under any data distribution, Bonferroni correction is unnecessary and the family-wise error rate (FWER) is equal to the Type I error of the individual tests, as formalized below.

Theorem 3. *For any auditor strategy π, the dual testing procedure with stopping thresholds $1/\alpha$ provides safe anytime-valid control of the FWER at level α.*

The main hyperparameter of the dual testing procedure is the choice of the auditor's budget m, where a higher value means a more stringent audit is conducted. This should be selected based on prior knowledge and real-world constraints. One should generally choose a higher m to mitigate failure modes in high-risk settings or find failure modes with lower prevalence. If the goal is to ensure a typical user is unlikely to encounter a failure mode, m can also be chosen to approximate how many test cases a typical user may conduct to assess the robustness of an AI system. That said, our ablation study (in the Appendix) shows that the procedure is relatively robust to the choice of m. More detailed guidance for hyperparameter tuning can be found in the Appendix.

4 Experiments

We evaluate the dual anytime-valid testing framework in two experimental settings. Experiment 1 uses semi-synthetic data to comprehensively test a variety of settings that vary the size and magnitude of the failure mode. Experiment 2 evaluates the practical utility of the methods on a real-world LLM pipeline for clinical note analysis. Failure modes are defined as poor-performing subgroups with prevalence ϵ at least 0.05. All hypothesis tests were conducted at level $\alpha = 0.05$.

We test several auditing strategies and testing procedures, with the primary goal of comparing the power of the testing procedures given the same auditing strategy. First, we apply the simplest `LR` e-process formulation with three different fixed auditing strategies: `Stratified` (uniform sampling across known subgroups), `Pre-learned` (pre-learn a model for sampling, based on the batch approach as in [Feng et al., 2023]), and `Oracle` (perfect subgroup knowledge in Experiment 1, which upper bounds performance) or `Informed Prior` (informed sampling of a known failure mode in Experiment 2, though not necessarily optimal). Next, we test the four e-processes from Section 3.2 (`LR`, `LR-UI`, `SR-LR`, and `SR-LR-UI`) with an adaptive auditing strategy based on online learning. This adaptive auditor converts LLM queries (X_t) into embeddings and retrains an NN ensemble with each additional annotation. Based on the predicted scores, the auditor samples from the ϵ-sized subgroup with the lowest predicted scores below the threshold q, if such a subgroup exists; otherwise, based on the Upper Confidence Bound (UCB) algorithm [Auer et al., 2002], it samples the bottom half of the population with the lowest confidence bounds for the estimated performance. The adaptive alternative probabilities in `LR-UI` and `SR-LR-UI` are constructed using the Exponentially Weighted Averaging Forecaster (EWAF) algorithm [Cesa-Bianchi and Lugosi, 2006]. Note that for all experiments, the auditor does not have direct access to strata information, unless specified otherwise; in doing so, our goal is to mimic real-world settings where strata information is often unavailable.

In the Appendix, we provide full experimental details, demonstrate that the test procedures are robust to the choice of budget m in ablation studies, and show results with similar trends on the CUB-200-2011 image dataset [Wah et al., 2011].

4.1 Experiment 1: Detecting failure modes with semi-synthetic data

To comprehensively evaluate testing methods across a variety of conditions, we simulated an AI pipeline with varying degrees of degradations in the GPQA dataset [Rein et al., 2023]. Following the available strata of science domains (Biology, Chemistry, Physics) and education levels (Undergraduate, Graduate, Post-graduate), we simulated a globally robust model with no failure modes (*None*) and models with a failure mode in the Chemistry domain, particularly for higher education levels (*Small*, *Medium* and *Large degradation*). We conduct experiments across two settings commonly found in practice: (a) the auditor has access to an unlabeled dataset from which it can actively choose observations to label, and (b) the auditor has no pre-existing data and must actively generate test cases. We simulate the latter by generating data from an LLM, and then use the auditor's accuracy model to prioritize which observations to annotate; see Appendix B.2 for a more detailed discussion on the interpretation of such an auditor. We evaluate each method over 100 simulation replicates, measuring rejection rates of $H_0^{\texttt{mod}}$ (power to detect failure modes) and $H_0^{\texttt{aud}}$ (audit passage), as well as the number of observations required to reach a decision.

Among the adaptive testing methods, `SR-LR-UI` achieved the highest power for detecting failure modes (Fig 2). UI variants (`SR-LR-UI`, `LR-UI`) generally outperformed their fixed-probability counterparts (`SR-LR`, `LR`), as using a probability estimator can further optimize the auditor's bets and thus power for detecting failure modes. Integrating changepoint detection also improves power for UI procedures (i.e., `SR-LR-UI` $>$ `LR-UI`), though it had a minimal effect for fixed-probability methods. The reason is that in the setting where the auditor can adaptively place bets against the failure rate, `SR-LR-UI` can tune the alternative probabilities to match the failure mode identified by the auditor after some initial learning period, while `LR-UI` is constrained to finding alternative probabilities that work well from the very beginning. In contrast, in the fixed-probability setting, the two methods are more similar because the alternative probabilities are fixed to be the same, leaving little room for optimizing power.

As expected, adaptive testing strategies substantially outperformed the pre-specified auditing approaches of `Stratified` and `Pre-learned` sampling. Furthermore, because the pre-specified auditing strategies have low power to uncover failure modes, their tests frequently reject the auditor's

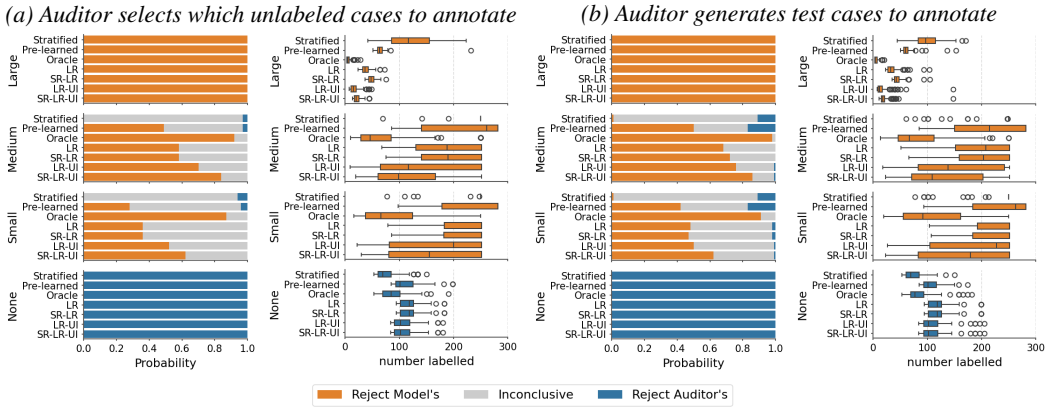


Figure 2: Detecting failure modes with semi-synthetic data (Experiment 1). The rows correspond to settings where the model has a failure mode with decreasing magnitude/prevalence of the degradation (*Large*, *Medium*, *Small*), where the bottom corresponds to *No failure mode*. Left column: rate of rejecting dueling null hypotheses, where orange indicates rate of detecting a failure mode, blue indicates passing the audit, and gray corresponds to neither (inconclusive). Right column: boxplots of the number of cases annotated to reach the correct conclusion or the maximum sample size.

null. While this is mathematically correct since the AI system has passed these specific audits, it risks being misinterpreted as indicating robustness of the AI system. Finally, the `Oracle` auditor outperforms all the existing methods since it knows the true failure mode, showing that rejection can be as fast as 15–25 observations. This shows that if one has an even smarter auditing strategy, such as through prior knowledge, one may be able to complete the statistical test with even smaller test suites.

When the model's null is true, every method correctly rejects $H_0^{\texttt{aud}}$ at high rates, certifying that the audit has passed. This confirms that Type I error for $H_0^{\texttt{mod}}$ is controlled: no method falsely reports a failure mode when none exists. We highlight that substantially more observations are needed by the testing procedures that use more adaptive auditors as they are better at finding poor-performing subgroups. This is desirable, as it means the auditing strategy was indeed quite good and a lot more data was needed to confidently reject the auditor's null.

4.2 Experiment 2: Auditing an LLM pipeline for clinical notes

We next applied our framework to a real-world task: auditing an LLM-based pipeline originally developed by Kothari et al. [2026] that extracts social determinants of health (SDoH) from clinical notes across 28 categories, including housing stability, substance use, mental health, safety concerns, and patient support networks. The original dataset includes human annotations across 81 clinical notes for all categories, yielding 2268 total comparisons. The overall LLM-human agreement rate was 88%, but category-level accuracy varied from 58% to 100%. To evaluate the proposed auditing procedures on this dataset, we set the auditing threshold to $q = 0.83$.

Figure 3 reveals similar qualitative patterns in real data, where `SR-LR-UI` achieves the highest power and requires the fewest samples for rejecting the model's null (except for `Informed Prior` which knows which SDoH categories perform poorly). Visualizing which SDoH categories are being selected by the adaptive auditor, we also find that the adaptive auditor indeed learns from just query embeddings to concentrate on the low-accuracy categories (Figure 3 top left), with sampling frequency strongly negatively correlated with the category's accuracy (Spearman $\rho = -0.84$).

5 Discussion

To understand the statistical underpinnings of highly flexible and targeted evaluation of AI workflows, this work introduces a formal hypothesis testing framework with procedures that provide anytime-valid guarantees under arbitrary adaptive sampling and optional stopping. The key insight is to represent robustness auditing of AI systems from two dueling perspectives: the model's null that asserts there are no failure modes, and the auditor's null that asserts they will find one. Then using the

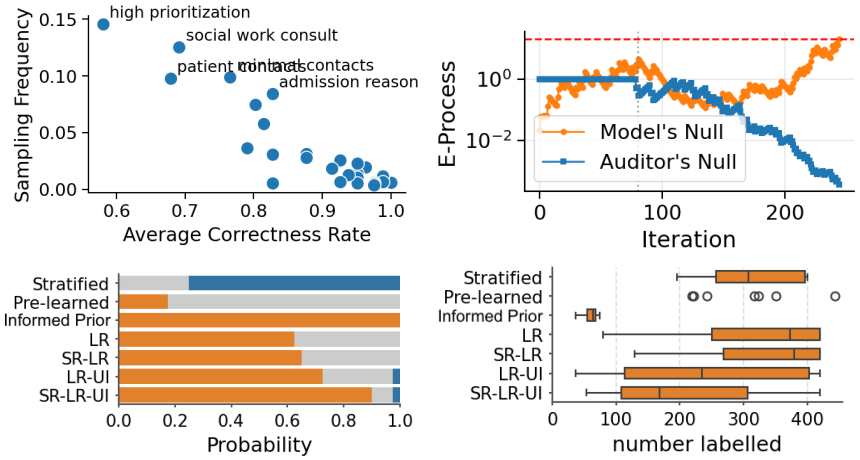


Figure 3: Auditing an LLM pipeline for extraction of Social Determinants of Health (SDoH) from clinical notes (Experiment 2). Top left: Sampling frequency vs. accuracy of different categories of SDoH clinical notes for `SR-LR-UI`, showing that the auditor learns to upsample low-accuracy categories. Top right: Example dual e-process with orange and blue curves representing the tests for the model's and auditor's null, respectively. The test detects that the AI system is not robust, as the orange curve exceeds the threshold of $1/\alpha$ (red dashed line). Bottom left: Rates at which the AI system fails the robustness audit (orange) versus passage of the audit (blue); gray is inconclusive. Bottom right: Boxplots of the number of clinical notes annotated until robustness is rejected or the maximum sample size is reached.

testing-by-betting framework, we introduce e-process-based procedures that maintain finite-sample Type I error control regardless of how adaptively the test cases are selected. Code for reproducing all results is available at `https://github.com/jjfenglab/safe-adaptive-testing`.

Future directions of this work include measuring and optimizing auditor quality; while this work provides a statistical foundation for any black-box auditor, the reliability of its outputs depends on the auditor's strength. Also, our empirical evaluations used an automated, albeit unpredictable, auditor to comprehensively test the statistical procedures across a variety of settings. Future work should study the suitability of this framework with not only more sophisticated automatic auditors but also human-in-the-loop systems where humans and AI agents collaborate to find failure modes (see e.g., ProbeLLM [Huang et al., 2026] or Domino [Eyuboglu et al., 2023]). Ultimately, by providing a general statistical framework for common auditing practices of modern AI systems, we hope to empower practitioners to confidently assess AI robustness without sacrificing statistical validity.

Acknowledgments

The authors thank Jialin Ouyang, Hanyang Li, Avni Kothari, and Roman Kouznetsov for helpful discussions on this project. This work was supported by the Weill Cancer Hub West and the Weill Family Foundation.

References

Martin Anthony and Peter L Bartlett. *Neural network learning: theoretical foundations*. Cambridge University Press, Cambridge, 1999.

Anthropic. Demystifying evals for AI agents. `https://www.anthropic.com/engineering/demystifying-evals-for-ai-agents`, January 2026. Accessed: 2026-4-28.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47(2-3):235–256, May 2002.

Elron Bandel, Asaf Yehudai, Lilach Eden, Yehoshua Sagron, Yotam Perlitz, Elad Venezian, Natalia Razinkov, Natan Ergas, Shlomit Shachor Ifergan, Segev Shlomov, Michal Jacovi, Leshem Choshen,

Liat Ein-Dor, Yoav Katz, and Michal Shmueli-Scheuer. General agent evaluation. *arXiv [cs.AI]*, February 2026.

Gabrielle Berrada, Jannik Kossen, Freddie Bickford Smith, Muhammed Razzak, Yarin Gal, and Tom Rainforth. Scaling up active testing to large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, October 2025.

Gérard Biau and Benoît Cadre. Optimization by gradient boosting. In *Advances in Contemporary Statistics and Econometrics*, pages 23–44. Springer International Publishing, Cham, 2021.

Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie Morgenstern, and Duen Horng Chau. FAIRVIS: Visual analytics for discovering intersectional bias in machine learning. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 46–56, October 2019.

Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, March 2006.

Alexandra Chouldechova and Aaron Roth. A snapshot of the frontiers of fairness in machine learning. *Commun. ACM*, 63(5):82–89, April 2020.

Ronald Christensen. Testing fisher, neyman, pearson, and bayes. *Am. Stat.*, 59(2):121–126, May 2005.

Ben Chugg, Santiago Cortes-Gomez, Bryan Wilder, and Aaditya Ramdas. Auditing fairness by betting. In *Advances in Neural Information Processing Systems 36*, pages 6070–6091, San Diego, California, USA, 2023. Neural Information Processing Systems Foundation, Inc. (NeurIPS).

Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. Slice finder: Automated data slicing for model validation. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 1550–1553, April 2019.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. SPECTER: Document-level representation learning using citation-informed transformers. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Stroudsburg, PA, USA, 2020. Association for Computational Linguistics.

Sabri Eyuboglu, Maya Varma, Khaled Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Ré. Domino: Discovering systematic errors with cross-modal embeddings. *International Conference on Learning Representations*, 2023.

Michael Feffer, Anusha Sinha, Zachary C Lipton, and Hoda Heidari. Red-teaming for generative AI: Silver bullet or security theater? *arXiv [cs.CY]*, January 2024.

Jean Feng, Alexej Gossmann, Romain Pirracchio, Nicholas Petrick, Gene Pennello, and Berkman Sahiner. Is this model reliable for everyone? testing for strong calibration. *AISTATS*, July 2023.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, Andy Jones, Sam Bowman, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Nelson Elhage, Sheer El-Showk, Stanislav Fort, Zac Hatfield-Dodds, Tom Henighan, Danny Hernandez, Tristan Hume, Josh Jacobson, Scott Johnston, Shauna Kravec, Catherine Olsson, Sam Ringer, Eli Tran-Johnson, Dario Amodei, Tom Brown, Nicholas Joseph, Sam McCandlish, Chris Olah, Jared Kaplan, and Jack Clark. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv [cs.CL]*, August 2022.

Irena Gao, Gabriel Ilharco, Scott Lundberg, and Marco Tulio Ribeiro. Adaptive testing of computer vision models. *IEEE/CVF International Conference on Computer Vision*, 2023.

Avijit Ghosh, Yifan Mai, Georgia Channing, and Leshem Choshen. AI evals are becoming the new compute bottleneck. *EvalEval Coalition Blog*, 2026.

Jack Goetz, Ambuj Tewari, and P Zimmerman. Active learning for non-parametric regression using purely random trees. *Neural Inf Process Syst*, 31:2542–2551, 2018.

Peter Grünwald, Rianne de Heide, and Wouter Koolen. Safe testing. *J. R. Stat. Soc. Series B Stat. Methodol.*, 86(5):1091–1128, November 2024.

David Hartmann, Lena Pohlmann, Lelia Hanslik, Noah Gießing, Bettina Berendt, and Pieter Delobelle. Audit me if you can: Query-efficient active fairness auditing of black-box LLMs. *arXiv [cs.LG]*, January 2026.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv [cs.CV]*, December 2015.

Ursula Hebert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (Computationally-identifiable) masses. *International Conference on Machine Learning*, 80:1939–1948, 2018.

Yue Huang, Zhengzhe Jiang, Yuchen Ma, Yu Jiang, Xiangqi Wang, Yujun Zhou, Yuexing Hao, Kehan Guo, Pin-Yu Chen, Stefan Feuerriegel, and Xiangliang Zhang. ProbeLLM: Automating principled diagnosis of LLM failures. *arXiv [cs.CL]*, February 2026.

Hamel Husain and Shreya Shankar. LLM evals: Everything you need to know – hamel's blog - hamel husain. `https://hamel.dev/blog/posts/evals-faq/`, January 2026. Accessed: 2026-1-17.

Avni Kothari, Patrick Vossler, Jean Digitale, Mohammad Forouzannia, Elise Rosenberg, Michele Lee, Jennee Bryant, Melanie Molina, James Marks, Lucas Zier, and Jean Feng. When the domain expert has no time and the LLM developer has no clinical expertise: Real-world lessons from LLM co-design in a safety-net hospital. *Proc. Conf. AAAI Artif. Intell.*, 2026.

Qi Kuang, Bowen Gang, and Yin Xia. Active hypothesis testing under computational budgets with applications to GWAS and LLM. *arXiv [stat.ME]*, December 2025.

Deokjae Lee, Junyeong Lee, Jung-Woo Ha, Jin-Hwa Kim, Sang-Woo Lee, Hwaran Lee, and Hyun Oh Song. Query-efficient black-box red teaming via bayesian optimization. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11551–11574, Toronto, Canada, July 2023. Association for Computational Linguistics.

Xiang Lisa Li, Farzaan Kaiyom, Evan Zheran Liu, Yifan Mai, Percy Liang, and Tatsunori Hashimoto. AutoBencher: Towards declarative benchmark construction. In *The Thirteenth International Conference on Learning Representations*, October 2024.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, February 2023.

Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. Algorithmic fairness: Choices, assumptions, and definitions. *Annu. Rev. Stat. Appl.*, 8(1):141–163, March 2021.

Peter C O'Brien and Thomas R Fleming. A multiple testing procedure for clinical trials. *Biometrics*, 35(3):549–556, 1979.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3419–3448, Stroudsburg, PA, USA, 2022. Association for Computational Linguistics.

Stuart J Pocock. Group sequential methods in the design and analysis of clinical trials. *Biometrika*, 64(2):191–199, August 1977.

Aaditya Ramdas and Ruodu Wang. Hypothesis testing with e-values. *arXiv [math.ST]*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. GPQA: A graduate-level google-proof Q&A benchmark. *arXiv [cs.AI]*, November 2023.

Marco Tulio Ribeiro and Scott Lundberg. Adaptive testing and debugging of NLP models. In *ACL 2022*, May 2022.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Stroudsburg, PA, USA, 2020. Association for Computational Linguistics.

S W Roberts. A comparison of some control chart procedures. *Technometrics*, 8(3):411–430, 1966.

Shuvom Sadhuka, Drew Prinster, Clara Fannjiang, Gabriele Scalia, Aviv Regev, and Hanchen Wang. E-valuator: Reliable agent verifiers with sequential hypothesis testing. *arXiv [cs.LG]*, December 2025.

Erwan Scornet, Gérard Biau, and Jean-Philippe Vert. Consistency of random forests. *Ann. Stat.*, 43 (4):1716–1741, August 2015.

Glenn Shafer. Testing by betting: A strategy for statistical and scientific communication. *J. R. Stat. Soc. Ser. A Stat. Soc.*, 184(2):407–431, April 2021.

Jaehyeok Shin, Aaditya Ramdas, and Alessandro Rinaldo. E-detectors: a nonparametric framework for sequential change detection. *New England Journal of Statistics in Data Science*, 2024.

A N Shiryaev. On optimum methods in quickest detection problems. *Theory Probab. Appl.*, 8(1): 22–46, January 1963.

Harvineet Singh, Fan Xia, Mi-Ok Kim, Romain Pirracchio, Rumi Chunara, and Jean Feng. A brief tutorial on sample size calculations for fairness audits. *Workshop on Regulatable Machine Learning at the 37th Conference on Neural Information Processing Systems*, December 2023.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv [cs.LG]*, December 2009.

Adarsh Subbaswamy, Roy Adams, and Suchi Saria. Evaluating model robustness and stability to dataset shift. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2611–2619. PMLR, 2021.

Adarsh Subbaswamy, Berkman Sahiner, Nicholas Petrick, Vinay Pai, Roy Adams, Matthew C Diamond, and Suchi Saria. A data-driven framework for identifying patient subgroups on which an AI/machine learning model may underperform. *NPJ Digit. Med.*, 7(1):334, November 2024.

U.S. Food and Drug Administration. Recommended content and format of non-clinical bench performance testing information in premarket submissions. `https://www.fda.gov/regulatory-information/search-fda-guidance-documents/recommended-content-and-format-non-clinical-bench-performance-testing-information-premarket`, October 2020. Accessed: 2026-4-14.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1576–1601, Stroudsburg, PA, USA, 2024. Association for Computational Linguistics.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-UCSD birds-200-2011 dataset. https://authors.library.caltech.edu/records/cvm3y-5hh21, 2011. Accessed: –.

A Wald. Sequential tests of statistical hypotheses. *Ann. Math. Stat.*, 16(2):117–186, June 1945.

Larry Wasserman, Aaditya Ramdas, and Sivaraman Balakrishnan. Universal inference. *Proc. Natl. Acad. Sci. U. S. A.*, 117(29):16880–16890, July 2020.

Eugene Yan. Product evals in three simple steps. https://eugeneyan.com/writing/product-evals/, November 2025. Accessed: 2026-4-28.

Tom Yan and Chicheng Zhang. Active fairness auditing. In *International Conference on Machine Learning*, pages 24929–24962. PMLR, June 2022.

Bin Yu and Karl Kumbier. Veridical data science. *Proc. Natl. Acad. Sci. U. S. A.*, 117(8):3920–3929, February 2020.

Zhiyuan Zeng, Yizhong Wang, Hannaneh Hajishirzi, and Pang Wei Koh. EvalTree: Profiling language model weaknesses via hierarchical capability trees. *Conference on Language Modeling*, 2026.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. *Proc. Int. Conf. Mach. Learn.*, pages 11492–11502, 2020.

Siyu Zhou, Jialin Ouyang, Fan Xia, Adarsh Subbaswamy, Drew Prinster, Suchi Saria, Augusto Garcia-Agundez, Sarah Pollet, Jinoos Yazdany, Patrick Vossler, and Jean Feng. Learning to detect shifts in unstructured data streams via monitoring-by-betting. *arXiv [cs.LG]*, October 2026.

Yinglun Zhu and Robert Nowak. Active learning with neural networks: Insights from nonparametric statistics. In S Koyejo, S Mohamed, A Agarwal, D Belgrave, K Cho, and A Oh, editors, *Advances in Neural Information Processing Systems 35*, volume 35, pages 142–155, San Diego, California, USA, 2022. Neural Information Processing Systems Foundation, Inc. (NeurIPS).

A Proofs

Proof of Theorem 1. Recall the model's hypotheses are

$$\begin{aligned} H_0^{\texttt{mod}}: &\quad \forall S \in \mathcal{S}_\epsilon, \quad V(S) \geq q, \\ H_1^{\texttt{mod}}: &\quad \exists S \in \mathcal{S}_\epsilon \text{ such that } V(S) < q. \end{aligned}$$

and the auditor's hypotheses can be formulated as:

$$\begin{aligned} &H_0^{\texttt{aud},[m]} : \exists \tau \in [m] \text{ such that } \forall t \geq \tau,\ V(S_t^\pi) < q, \\ &H_1^{\texttt{aud},[m]} : \forall \tau \in [m], \exists t \geq \tau \text{ such that } \ V(S_t^\pi) \geq q. \end{aligned}$$

As $m \to \infty$, denote

$$H_0^{\texttt{aud},\mathbb{N}} := \lim_{m\to\infty} H_0^{\texttt{aud},[m]}: \quad \exists \tau \in \mathbb{N} \text{ such that } \forall t \geq \tau, V(S_t^\pi) < q \tag{9}$$

$$H_1^{\texttt{aud},\mathbb{N}} := \lim_{m\to\infty} H_1^{\texttt{aud},[m]}: \quad \forall \tau \in \mathbb{N},\ \exists t \geq \tau \text{ such that } V(S_t^\pi) \geq q. \tag{10}$$

The limits are well defined since both $H_0^{\texttt{aud},[m]}$ and $H_1^{\texttt{aud},[m]}$ are monotone in m.

Since $H_0^{\texttt{mod}}$ and $H_1^{\texttt{aud},\mathbb{N}}$ are the respective complements of $H_1^{\texttt{mod}}$ and $H_0^{\texttt{aud},\mathbb{N}}$, we only need to prove $H_1^{\texttt{mod}} \Leftrightarrow H_0^{\texttt{aud},\mathbb{N}}$.

Notice that if $H_0^{\texttt{aud},\mathbb{N}}$ is true, then there exists a large T such that $V(S_T^\pi) < q$ and $H_1^{\texttt{mod}}$ is true by taking $S^* = S_T^\pi$. On the other hand, if $H_1^{\texttt{mod}}$ is true, then by asymptotic consistency of the auditor's strategy, $\lim_{t\to\infty} V(S_t^\pi) < q$ and thus $H_0^{\texttt{aud},\mathbb{N}}$ is true. Therefore, $H_1^{\texttt{mod}}$ is true iff $H_0^{\texttt{aud},\mathbb{N}}$.

□

Proof of Theorem 2. For any $m' \in [m]$, let $S^{1,\pi} = \left\{S_1^{1,\pi}, S_2^{1,\pi}, \ldots, \right\}$ be any sequence of subsets which are identified by the auditor's sampling strategy π and satisfy $H_1^{\texttt{aud},m'}$. Suppose, to the contrary, that $\sup_{t<m'} \mathbb{E}(\phi(t); S_1^{1,\pi}, \ldots, S_t^{1,\pi}) > \alpha$, i.e., the power can be above α at some $t < m'$.

Next, construct $S^{0,\pi} = \left\{S_1^{0,\pi}, S_2^{0,\pi}, \ldots, \right\}$ such that $S^{0,\pi}$ satisfies $H_0^{\texttt{aud},m'}$ and $S_t^{0,\pi} = S_t^{1,\pi}$ for $t < m'$. However, this implies

$$\sup_{t<m'} \mathbb{E}(\phi(t); S_1^{0,\pi}, \ldots, S_t^{0,\pi}) = \sup_{t<m'} \mathbb{E}(\phi(t); S_1^{1,\pi}, \ldots, S_t^{1,\pi}) > \alpha,$$

which contradicts $\sup_t \mathbb{E}\left(\phi(t)|H_0^{\texttt{aud},m'} \text{ is true}\right) \leq \alpha$. Therefore, by contradiction, $\sup_{t<m'} \mathbb{E}\left(\phi(t)|H_1^{\texttt{aud},m'} \text{ is true}\right) \leq \alpha$, which holds for any $m' \in [m]$.

□

Proof of Theorem 3. If the model's null is true, then the auditor's null cannot be true. If the auditor's null is true, then the model's null cannot be true. Since only one of these null hypotheses can be true, then if both of the individual tests are controlled at level α, the FWER is controlled at level α. □

B Testing procedures

B.1 Pseudocode for dual testing procedure

The pseudocode for dual testing is given in Algorithm 1. The model's test $E_t^{\texttt{mod}}$ can be $E_t^{\text{LR},\texttt{mod}}$, $E_t^{\text{LR-UI},\texttt{mod}}$, $E_t^{\text{SR-LR},\texttt{mod}}$ or $E_t^{\text{SR-LR-UI},\texttt{mod}}$ and the auditor's test $E_t^{\texttt{aud}}$ can be $E_t^{\text{LR},\texttt{aud}}$ or $E_t^{\text{LR-UI},\texttt{aud}}$.

Algorithm 1 Dual testing

Initialize $E_0^{\texttt{mod}} = E_0^{\texttt{aud}} = 1$, an auditing strategy π.
for $t = 1, 2, \ldots$ **do**
  Pick subgroup S_t^π based on the audit strategy π.
  Sample test observation X_t from S_t^π.
  Evaluate the AI system's response on X_t and obtain the score Y_t.
  Update $E_t^{\texttt{mod}}$.
  if $E_t^{\texttt{mod}} > 1/\alpha$ **then**
    return Reject $H_0^{\texttt{mod}}$: Failure mode is detected.
  end if
  if $t \geq m$ **then**
    Update $E_t^{\texttt{aud}}$.
    if $E_t^{\texttt{aud}} > 1/\alpha$ **then**
      return Reject $H_0^{\texttt{aud},m}$: Audit is passed.
    end if
  end if
end for

B.2 Auditing strategies

With rigorous finite-sample inference being possible under active testing based on the theoretical results above, the practical value of this framework depends on having a good auditing strategy π for discovering failure modes.

Scenario (a): Unlabeled data. If one has unlabeled data from the target population, one could learn a poor-performing subgroup with online learning. For instance, one may (re)train an AI/ML model $\hat{\pi}$ at the start of each iteration t on labeled data collected up to time $t-1$ to estimate the conditional expectation of the score $E[Y|X]$. This can then be used to determine sampling. An appropriate exploitation-exploration tradeoff should be made, to ensure asymptotic consistency of the auditor. In our experiments, the auditor greedily defines the subset S_t^π as the observations X predicted to be

in the bottom ϵ percentile *and* have predicted scores $\hat{\pi}(X)$ below the target threshold q. If such an ϵ-sized subgroup does not exist, then the auditor favors exploration at time t instead by sampling among observations whose lower confidence bound of the predicted score falls within the bottom 50%, following the UCB strategy in the bandit literature [Auer et al., 2002].

Scenario (b): Data generation. In certain settings, one may not have a pre-existing unlabeled dataset $\{X_j\}$. This can occur, for instance, when one is building a new AI pipeline, where no data has yet been collected from the target application. To audit such an AI pipeline, the auditor must generate X values to audit instead. In the ideal setup, the auditor can (i) determine that its selected subgroup S_t^π has prevalence at least ϵ and (ii) generate observations from S_t^π with respect to the probability measure μ restricted to the subgroup. While the former is plausible as long as the auditor has adequate prior knowledge, the latter is highly dependent on the auditor's generation capabilities in practice (both in the setting where X is constructed manually or automatically using a foundation model). In such cases, an accurate and more honest description of the audit should be with respect to the actual data generation mechanism, which is an approximation of the probability measure in the target application. For instance, if a failure mode is detected under an approximate generator, it means that the auditor has found some subgroup that the AI system performs poorly for, though this subgroup may have smaller prevalence than the target threshold of ϵ. If the auditor using the generator failed to reject the model's null hypothesis (i.e., there is no failure mode), it means that the auditor failed to find a failure mode for the generator's weighted (or biased) view of the target population, not necessarily that there is no failure mode globally. The interpretation of the auditor's null should be adjusted similarly.

B.3 Practical guide for hyperparameters

Stopping rule. The test stops when the e-statistic exceeds $1/\alpha$, where α is the desired Type I error rate. The user only needs to specify α, where the most typical choice is 0.05.

Hyperparameters defining the hypothesis test. Prevalence threshold ϵ and performance threshold q define what a failure mode is and what the null hypothesis is. Budget m determines how many observations the auditor can explore prior to the auditor's null being tested, and defines the auditor's null hypothesis. All three should be chosen based on prior knowledge and the goal of the auditor. For instance, one may choose q and ϵ based on regulatory or model fairness requirements. Budget m may be chosen to guard against detection of a failure mode by a typical user testing out the LLM application (so set m to the typical number of attempts) or by some function class (so set m to the typical training sample size needed). We think these hyperparameters can often be chosen based on prior knowledge and by working closely with domain experts. Nevertheless, in the absence of any prior knowledge, one could test a range of values and check if results remain robust, which can be viewed as a sensitivity or stability analysis (such as from veridical data science [Yu and Kumbier, 2020]). For example, ablation studies in Section C of the Appendix show that the relative performance of different methods is robust to the choice of m.

Auditor hyperparameters. We recommend tuning auditor hyperparameters in an online approach, such as the Exponentially Weighted Averaging Forecaster (EWAF) algorithm [Cesa-Bianchi and Lugosi, 2006].

C Experiment 1: Detailed Setup

We provide complete details for the semi-synthetic experiments described in Section 4.

Data and label simulation. We use the GPQA dataset [Rein et al., 2023] which contains 448 graduate-level multiple-choice science questions spanning three domains: Chemistry (187 questions), Biology (142 questions), and Physics (119 questions). Each question is also categorized by education level: easy undergraduate, hard undergraduate, hard graduate, and post-graduate.

We simulate binary accuracy labels (correct/incorrect) by assigning each domain$\times$education level combination a ground-truth accuracy probability. As an illustration, Table 1 shows the accuracy probabilities in Scenario (a) discussed below for Chemistry (the poor-performing subgroup) across

education levels; Biology and Physics maintain accuracy ≥ 0.85 in all settings. The null threshold is $q = 0.85$ throughout.

Degradation	Chemistry Accuracy	Gap from q	Model's Null
Large	0.25–0.40	0.45–0.60	False
Medium	0.50–0.75	0.35–0.10	False
Small	0.60–0.77	0.25–0.08	False
None	0.92–0.96	> 0 (above q)	True

Table 1: Simulated accuracy settings for Chemistry across magnitude of the degradation in Experiment 1 Scenario (a). Biology and Physics domains maintain accuracy ≥ 0.85 in all settings.

Testing parameters. The maximum sample budget is 250 observations. We initialize with $n_{\text{init}} = 2$ randomly sampled observations before active selection begins. The `Pre-learned` sampler uses an additional 30 held-out observations for pre-training.

Adaptive sampling. The adaptive sampling trains a model of the accuracy to identify the subgroup most likely to be a failure mode. The accuracy model uses the text embedding of the GPQA question as input, generated from the `sentence-transformers/allenai-specter` model [Cohan et al., 2020]. Accuracy is modeled using an ensemble of neural networks with one hidden layer, tuned over the hyperparameters of hidden dimension $\in \{5, 10\}$ and number of epochs $\in \{2, 4, 8, 16\}$. At each iteration, the best hyperparameter was based on the one with the smallest exponentially weighted held-out loss, which allows the selection procedure to adapt to shifts in the optimal hyperparameter as the accuracy model improved over time. We use $\epsilon = 0.05$ for ϵ-greedy exploration.

For UI-based estimators, we construct adaptive alternative probabilities in `LR-UI` and `SR-LR-UI` using the EWAF algorithm. That is, the predicted probability for testing the model's null, $\hat{\gamma}_{t-1}^{\texttt{mod}}$, is an adaptively weighted average over a pre-specified grid of B plausible probabilities $\mathcal{Q}_1^{\texttt{mod}} = \{q_b \in (0, q) : b \in [B]\}$ under $H_1^{\texttt{mod}}$. In particular, $\hat{\gamma}_{t-1}^{\texttt{mod}} = \sum_{b=1}^{B} u_{t-1,b} q_b$ with the weights $u_{t-1,b}$ defined as

$$u_{t,b} = \frac{u_{t-1,b} \exp\left(\lambda \log\left(\frac{p(Y_t;q_b)}{p(Y_t;q)}\right)\right)}{\sum_{b'=1}^{B} u_{t-1,b'} \exp\left(\lambda \log\left(\frac{p(Y_t;q'_b)}{p(Y_t;q)}\right)\right)}.$$

The predicted probability for testing the auditor's null, $\hat{\gamma}_{t-1}^{\texttt{aud}}$, is computed similarly.

Scenario (a): Unlabeled data. The auditor has access to all 448 GPQA questions without labels. At each iteration, the auditor selects one question to label based on predicted accuracy (greedy selection of lowest predicted accuracy, with ϵ-greedy exploration). The selected question is then removed from the pool.

Scenario (b): Data generation. The auditor generates a question from each of the 3×3 categories corresponding to the levels of domain and education level. (To control computational costs, we pre-generate a question bank using an LLM for each category, rather than querying an LLM at every iteration.) The auditor then selects among the generated candidates using ϵ-greedy exploration.

Evaluation. We run 100 simulation replicates for each simulation setting. For each replicate, we record: (1) which null hypothesis was rejected ($H_0^{\texttt{mod}}$ or $H_0^{\texttt{aud}}$); (2) the stopping time (number of observations until correct rejection or the maximum sample size is reached). We report rejection rates and boxplots of stopping times across trials.

Compute time. The procedure is quick to run and requires minimal infrastructure. With only two CPU cores with 5GB memory, a single audit completed within 15 minutes on average. Most of the time was spent updating the auditor with each new observation, as the auditor conducted hyperparameter tuning of the NN ensemble at each iteration. Significant speed-ups are possible by parallelizing hyperparameter tuning.

Ablation study on pre-auditor's budget m. We perform an ablation study to examine the effect of the budget m prior to starting the auditor's test on the performance of the proposed procedures. Specifically, we evaluate $m \in \{10, 40, 70\}$ while holding all other experimental settings constant. As shown in Figure C.1 for $m = 10$ and Figure C.2 for $m = 70$, results are qualitatively similar to those with $m = 40$ in Figure 2 in the main text. Thus, the procedure is relatively insensitive to the choice of budget m.

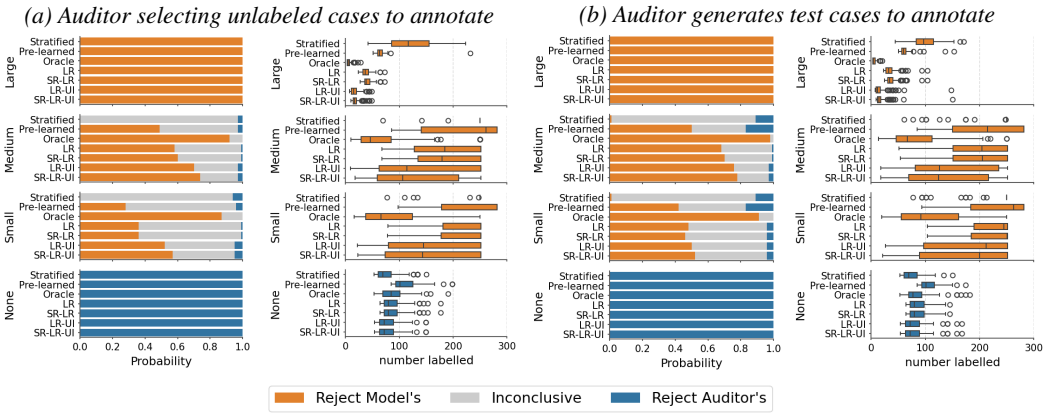


Figure C.1: Detecting failure modes with semi-synthetic data (Experiment 1) with $\mathbf{m = 10}$. The rows correspond to settings where the model has a failure mode with decreasing magnitude/prevalence of the degradation (*Large*, *Medium*, *Small*), where the bottom corresponds to *No failure mode*. Left column: rate of rejecting dueling null hypotheses, where orange indicates rate of detecting a failure mode, blue indicates passing the audit, and gray corresponds to neither (inconclusive). Right column: boxplots of the number of cases annotated during the audit to reach the correct conclusion or the maximum sample size.

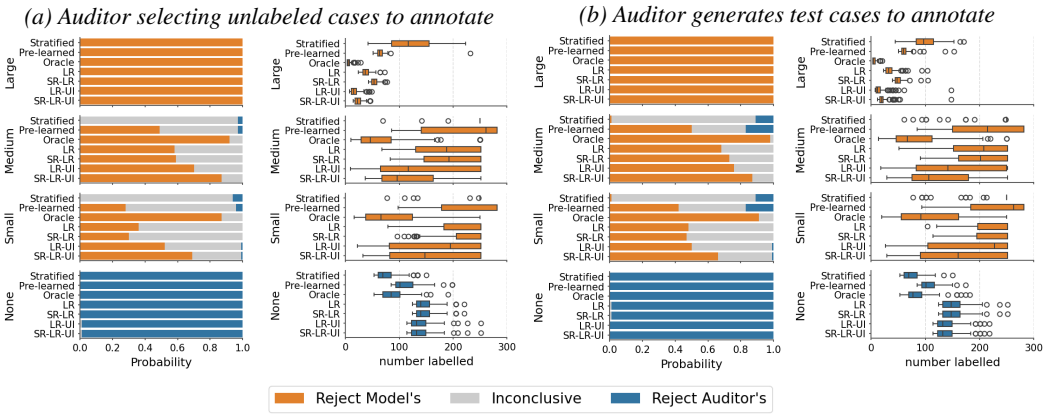


Figure C.2: Detecting failure modes with semi-synthetic data (Experiment 1) with $\mathbf{m = 70}$. The rows correspond to settings where the model has a failure mode with decreasing magnitude/prevalence of the degradation (*Large*, *Medium*, *Small*), where the bottom corresponds to *No failure mode*. Left column: rate of rejecting dueling null hypotheses, where orange indicates rate of detecting a failure mode, blue indicates passing the audit, and gray corresponds to neither (inconclusive). Right column: boxplots of the number of cases annotated during the audit to reach the correct conclusion or the maximum sample size.

D Experiment 2: Detailed Setup

We provide complete details for the clinical NLP experiment described in Section 4.

Data and task. We evaluate an LLM-based pipeline for extracting social determinants of health (SDoH) from clinical notes at an urban medical center. The pipeline processes discharge summaries

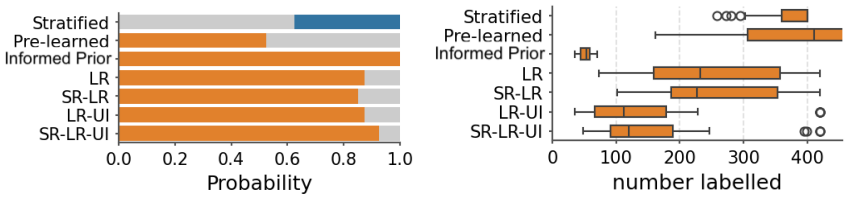


Figure D.3: Auditing an LLM pipeline for extraction of Social Determinants of Health (SDoH) from clinical notes (Experiment 2) with threshold $q = 0.85$. Left: Rates at which the AI system fails the robustness audit (orange) versus passage of the audit (blue); gray is inconclusive. Right: Boxplots of the number of clinical notes annotated until robustness is rejected or the maximum sample size is reached.

and social work notes to extract structured information across 28 SDoH categories, including housing stability, substance use patterns, mental health needs, safety concerns, patient contacts, and care coordination details. The dataset consists of 81 clinical notes with human expert annotations for each of the 28 categories, yielding 2,268 total note×category comparisons. Each comparison is scored as correct (1) if the LLM extraction matches the human annotation, and incorrect (0) otherwise. Table 2 provides the complete list of SDoH categories with their definitions as specified in the LLM prompts.

Category-level performance. The overall LLM-human agreement rate is 88%, but performance varies substantially across categories. The worst-performing categories include: high prioritization (58%), patient contacts (68%), social work consult (69%), minimal contacts (77%), and mental health (79%). The best-performing categories achieve 95–100% agreement, including substance use date of last use (100%), immigration interventions (99%), and several substance use subcategories. With the null threshold set to $q = 0.83$, the performance on 11 of the 28 categories falls below q.

Adaptive learning. For each note×category pair, we construct input features by concatenating (1) the category-specific prompt definition from Table 2 and (2) the clinical note text. We compute embeddings using the `sentence-transformers/allenai-specter` model [Cohan et al., 2020]. The accuracy model is the same neural network ensemble as that in Experiment 1.

Testing parameters. We set $m = 40$ (audit budget), a maximum sample size of 400, and $n_{\text{init}} = 20$ initial random samples. We report rejection rates and stopping times from 40 simulation replicates.

Additional results. We also carry out the experiment with a null threshold $q = 0.85$. As shown in Figure D.3, the comparison is similar to the case of $q = 0.83$ in the main text, with SR-LR-UI performing the best.

E More Results: CUB-200-2011 Image Data

Data and task. We evaluate an image classifier on the CUB-200-2011 dataset [Wah et al., 2011], which contains 11,788 images of 200 bird species. The model under audit is a ResNet-18 classifier. The auditor's goal is to determine whether there exists a subgroup whose accuracy falls below a null threshold $q = 0.85$. Binary accuracy labels are simulated by assigning each species a ground-truth accuracy probability, with white species receiving the highest accuracy, darker species moderate accuracy, and 18 black species the lowest.

Adaptive learning. We trained an accuracy model using the same NN ensemble architecture with embeddings extracted from a pre-trained ResNet-18 [He et al., 2015] as input features.

Testing parameters. We set the budget $m = 40$ prior to starting the auditor's test, a maximum sample budget of 250 observations, and $n_{\text{init}} = 20$ initial random observations before active selection begins. The `Pre-learned` sampler uses an additional 30 held-out observations for pre-training. We run 40 independent trials per method. Results are shown in Figure E.4.

Category	Definition
Reason for Admission	Identify the reason for the admission to the hospital.
Reason for SW Consult	For social worker consult note types only, identify the reason for the social work consult.
Mental Health	Identify any mention of mental health needs: depression, anxiety, suicidal thoughts, or psychiatric symptoms impacting functioning.
Mental Health Interventions	Extract actions/interventions for mental health needs, categorized as previous, current, or planned actions.
Safety	Identify any mention of safety needs: abuse, domestic violence, assault, interpersonal violence, recurrent falls, or self-neglect/caregiver neglect.
Safety Interventions	Extract actions/interventions for safety needs, categorized as previous, current, or planned actions.
Housing	Identify any mention of housing needs: homelessness, housing instability, unsafe living conditions, or inability to afford housing.
Housing Interventions	Extract actions/interventions for housing needs. Look for: home health, shelters, navigation centers, Medical Respite, SNF referrals, supportive housing.
Substance Use	Identify any mention of substance use needs: illegal drugs, prescription drug misuse, or problematic controlled substance use.
Substance Use Interventions	Extract actions/interventions for substance use needs. Look for: detox referrals, residential/outpatient SUD programs, Addiction Care Team.
Substance Use - OUD	Opioid Use Disorder: Identify if the patient has problematic opioid use (including fentanyl or heroin).
Substance Use - StUD	Stimulant Use Disorder: Identify if the patient has problematic stimulant use (amphetamines, cocaine, methamphetamine).
Substance Use - PSUD	Polysubstance Use Disorder: Identify if the note explicitly mentions PSUD.
Substance Use - MOUD	Medication for OUD: Identify if the patient is on methadone or buprenorphine.
Substance Use - Date of Last Use	If the patient has substance use history, identify the date of last use.
Food Insecurity	Identify any mention of food insecurity: inability to access adequate food due to financial constraints.
Food Insecurity Interventions	Extract actions/interventions for food insecurity. Look for: Meals on Wheels, Great Plates, Project Open Hand.
Immigration	Identify any mention of immigration needs: documentation concerns, deportation fears, language barriers.
Immigration Interventions	Extract actions/interventions for immigration needs.
Tobacco Use	Identify any mention of tobacco use: cigarettes, vaping, chewing tobacco, or other nicotine products.
Tobacco Use Interventions	Extract actions/interventions for tobacco use needs.
Alcohol Use	Identify any mention of alcohol use needs: excessive drinking, binge drinking, or alcohol-related problems.
Alcohol Use Interventions	Extract actions/interventions for alcohol use needs.
DME: Mobility Devices	Identify durable medical equipment the patient uses: wheelchairs, walkers, hospital beds, oxygen, CPAP/BiPAP, etc.
Patient Contacts	Identify references to patient's contacts: friends, family, case workers. Include name, phone number, and relation.
Lack of Social Support	Identify missing/minimal contacts or social isolation: widowed, lives alone, no family involved, no emergency contact.
Patient Independence	Identify inability to perform ADLs (bathing, dressing, toileting, eating) or IADLs (medications, transportation, cooking, finances).
High Priority	Identify if patient is high priority: homeless and discharge ready, Jane/John Doe, abuse/neglect/safety, need family located, death/dying, major life-changing event.
Outpatient Therapies	Identify home health services or SNF: PT/OT/SLP/RN/SW that come to patient's home, or IHSS caregivers.

Table 2: SDoH categories and their definitions as specified in the LLM extraction prompts. Definitions are abbreviated for space; full prompts include additional examples and classification guidance.

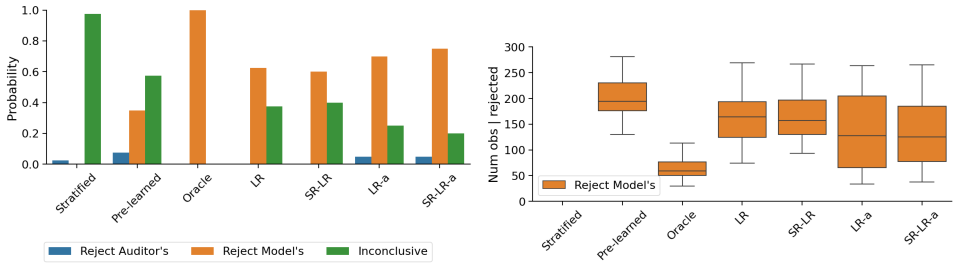


Figure E.4: Auditing a pre-trained ResNet-18 classifier for CUB-200-2011 bird classification. Left: Rates for rejection of robustness (orange) and passage of the audit (blue). Right: Boxplots of the number of images annotated when robustness is rejected or the maximum budget is exhausted.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the three main contributions: formalizing AI auditing as dueling hypothesis tests, introducing dual e-processes with SAVI guarantees, and empirical demonstration of Type-I error control. These claims are directly supported by Theorems 1 and 3 and the experimental results in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Discussion (Section 5) acknowledges key limitations, including that the framework treats auditing strategies as a black box and that empirical evaluations focus on automated auditors rather than human-in-the-loop settings.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.

- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theorems state their assumptions explicitly, and complete proofs are provided in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All experimental results can be reproduced using the code provided at the GitHub repository linked in the Discussion, which includes scripts for all experiments described in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed

instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code for all experiments is available at the GitHub repository linked in the Discussion. Data for all experiments in the paper are publicly available except for Experiment 2. Experiment 2 uses real-world clinical notes that cannot be shared publicly due to HIPAA restrictions, but may be obtained from the authors of Kothari et al. [2026] through an appropriate data sharing request.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

Justification: Section 4 specifies all experimental settings; full implementation details are provided in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper introduces methods for establishing statistical significance. In addition, all experiments were comprehensively evaluated over multiple replicates to quantify statistical uncertainty, which are summarized in boxplots in Figure 2 and Figure 3; additional results are also provided in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Details on compute resources used for all experiments are provided in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that this research conforms with it in every respect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The Discussion highlights how the method can empower practitioners to assess AI robustness in a statistically rigorous manner. The paper also warns against using weak auditing strategies, as they can be misinterpreted as certifying global robustness.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release any models or datasets that pose a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All owners of data assets have been cited. Usage of the datasets abided by their terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code released at the GitHub repository linked in the Discussion is documented with instructions for reproducing all experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: The paper analyzes clinical notes in Experiment 2, which constitutes research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: Appropriate IRB approval was obtained for analyzing the clinical notes in Experiment 2.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not the core component of the proposed methodology. They are only used to generate samples for auditing in Experiment 1b, which has been described in the Appendix.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.