From Method to Interface: Human-Centred Auditing Under Resource and Access Constraints

Anonymous Author(s)

Abstract

Large language models (LLMs) increasingly mediate consequential decisions through content moderation, screening, ranking, and risk scoring, yet independent evaluation of their harms remains structurally constrained. Systems are often only accessible via paid, rate-limited APIs; outputs are stochastic and prompt-sensitive; and model versions change without notice. What can be concluded about bias or harm is therefore not only a methodological question but a resource- and access-dependent one: independent auditors face high query costs and limited observability, while their findings are easily dismissed.

We argue that constrained evaluation is fundamentally an HCI problem of evidence production, involving choices about what to test, how to manage costs, how to communicate uncertainty, and how to turn partial observations into defensible claims across diverse auditor communities — from ML researchers to journalists and civil society practitioners. To address this, we present *Bounded Active Fairness Auditing* (BAFA), a query-efficient method that treats fairness evaluation as uncertainty estimation under strict budgets, reaching valid estimates with hundreds rather than thousands of black-box queries across two case studies. Building on BAFA and a review of structural constraints facing independent auditors, we derive ten design requirements for auditor-facing tools, covering budget transparency, metric selection guidance, live uncertainty display, defensible stopping, and collaboration support. We argue that developing semi-professional auditing interfaces that combine statistical rigour with accessibility is an important and underexplored direction for the HCI community.

CCS Concepts

• **Human-centered computing** → **Empirical studies in HCI**; • **Computing methodologies** → **Machine learning**; *Natural language processing*.

Keywords

LLMs, Evaluation Crisis, Auditing, LLM Evaluation, Accountability, Query-Efficiency, Active Auditing, Human-Centred Evaluation

ACM Reference Format:

Anonymous Author(s). 2018. From Method to Interface: Human-Centred Auditing Under Resource and Access Constraints. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 15 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

LLMs are now used not only for generating content but also in decision-making contexts such as content moderation, screening, ranking, and risk scoring. In these settings, concerns about bias and harm translate into material consequences for users and affected groups. Evaluation therefore becomes a central transparency and accountability question: what can be demonstrated about a deployed system, who is able to generate such evidence, and under what standards of proof it is recognised as valid. Because provider-led audits and internal evaluations are shaped by developers' priorities, incentives, and access conditions, independent evaluation is necessary to uncover harms that may remain invisible if evaluated by first parties [5, 24, 26, 33].

However, independent evaluation of LLM-based products is structurally constrained. Many systems are accessible only through paid APIs with strict rate limits and limited accessibility and observability of system components [9, 31, 32]. Outputs are stochastic, prompt- and context-sensitive, and can change with different versions, making it hard to reproduce results or trace causes [11, 18, 37]. Scientifically valid evaluation needs repeated testing with different prompts, baselines, seeds, and contexts [4, 19, 23, 30] and, thus, what can be concluded from an evaluation depends on resources like query budgets, computing power, tools, and system access [1, 3, 38]. Meanwhile, companies often have more such resources or even control them, whereas external auditors struggle with limited access and high costs [20, 25, 33]. As a result, there is an imbalance in evidence: independent findings are harder to produce and more likely to be dismissed as "scientifically invalid," even when they reveal real harms.

We argue that this is fundamentally an HCI problem: evaluation is not only a statistical exercise but a socio-technical practice of evidence production, interpretation, and justification under constraints. It involves choosing what to test, managing costs and risks, who to involve in the evaluation process, reverse-engineering with limited access, and clearly communicating uncertainty to support decision-making and actions. Prior HCI research shows that users, including marginalised groups, already engage in ad hoc testing and collective sensemaking to surface harmful algorithmic behavior [35], yet these practices remain time-intensive and weakly supported by tools [17]. While recent systems such as "WeAudit" begin to scaffold end-user auditing and translate findings into actionable insights [15], audit-tooling research indicates that the broader ecosystem still falls short of enabling accountability in practice [27]. We therefore argue that HCI should develop interfaces and strategies for constrained evaluation that lower query costs, visualise uncertainty, and help auditors turn partial and probabilistic evidence into defensible claims.

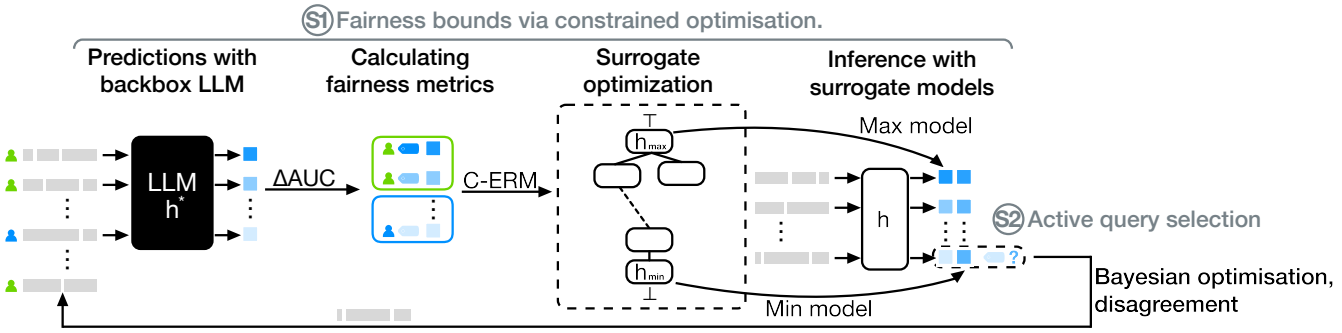


Figure 1: BAFA treats independent auditing as *uncertainty estimation* over a target metric under a strict query budget. In each round, BAFA (i) issues a small batch of black-box queries, (ii) computes a certificate interval $[\mu_{\min}, \mu_{\max}]$ for the fairness metric via constrained optimisation over a surrogate model family, and (iii) selects the next batch of queries to maximally shrink the interval (and thus the auditor's uncertainty).

Motivated by this gap, we present BAFA, a bounded active fairness auditing method that estimates group disparities with query efficiency. BAFA operationalises auditing as estimating uncertainty over a target metric and keeps a set of possible hypotheses that fit the observed outputs, calculating an interval for the fairness measure using constrained optimisation. Active query selection then targets areas that will most reduce this interval. The interval's width shows how much the fairness estimate could still change given the evidence so far, and it serves as an uncertainty signal for auditors to act on. We demonstrate work-in-progress empirical findings in two case studies in which BAFA needs up to 40x fewer queries than baselines and offer suggestions for uncertainty-aware interfaces using our method. Our contributions are: (1) introducing BAFA as an evaluation method for situations with limited access, (2) framing auditing as a process constrained by resources and access, and (3) offering an uncertainty-aware view of LLM evaluation that suggests new HCI directions for auditing tools, interfaces, and workflows.

2 Query-Efficient Auditing Tool: Bounded Active Fairness Auditing (BAFA)

2.1 Query-Efficient Auditing

Consider an independent auditor—a civil society group, journalist, or academic—who wants to evaluate whether a deployed (or API-provided) LLM system is unfair. The auditor typically has: *a dataset* that approximates a real-world input distribution (such as toxic comments or short biographies), including ground-truth labels and protected-group attributes for evaluation; *black-box access* to the system under audit that returns a score per input (such as toxicity probability or a confidence score); and *query limits* on how many inputs can be sent to the system as each query costs resources and has to be paid.

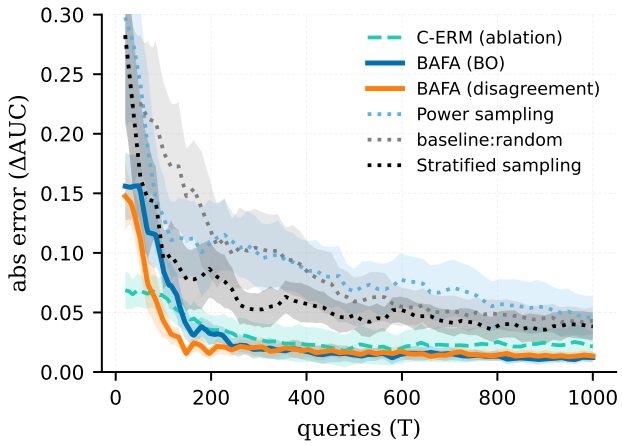
In many real-world evaluations, evaluation is constrained as each API call costs money, is rate-limited, may be logged, and can create privacy and safety risks when auditors must submit sensitive or stigmatising inputs [10]. In this setting, "better evaluation" is not primarily about achieving a slightly higher score on a benchmark, but it is about producing credible (quantitative) evidence about a system property (e.g., a group disparity when it comes to performance or outcome) with as few black-box queries as possible. The practical challenge is that fairness metrics (especially distributional, threshold-invariant ones) can require thousands of samples for stable estimates under naive sampling [36].

We use *query-efficient auditing* to mean that given a target quantity (such as aforementioned performance disparity) and a query budget T, design an interactive audit procedure that converges faster to a desired margin of error ϵ and provides usable uncertainty information during the audit so that auditors can decide when the current evidence is sufficient to act. This differs from red teaming and failure discovery, which aim to find severe counterexamples but do not estimate the prevalence or magnitude of a population-level metric [20].

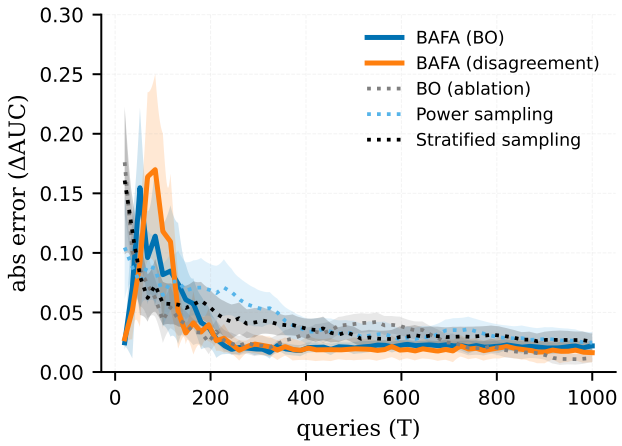
2.2 BAFA Workflow in Two Steps

BAFA operationalises constrained auditing as a sequential loop of producing evidence under uncertainty, designed for settings where auditors have limited access, limited budgets, and only API access. Figure 1 summarises the workflow:

S1 **Build an uncertainty-aware estimate** BAFA quantifies uncertainty about a model's group fairness by maintaining a set of surrogate hypotheses that remain compatible with the black-box scores observed so far. Concretely, we use a lightweight text model as a surrogate (a non-fine-tuned `bert-base-uncased`; [16]) and solve a constrained optimisation problem with Cooper [21]. In each audit round, we fit two extremal surrogate solutions that both agree with the black box on the queried examples (up to a small tolerance), but push the target fairness metric in opposite directions. These two feasible solutions define an interval of plausible fairness values given the currently available evidence. We treat the resulting interval width as an operational notion of uncertainty: wide intervals indicate that the observed queries still leave substantial ambiguity about the fairness



(a) Active auditing methods perform more query-efficient and stable over 20 Civil Comments seeds. BAFA methods (solid) converge significantly faster than baseline sampling strategies (dotted). Shaded areas indicate 95% confidence intervals across seeds and demonstrate that BAFA methods show substantially reduced variance compared to baseline methods.



(b) Active auditing methods perform well even with large parameter spaces with GPT-4.1-mini as black-box. Similarly, to Fig. a), BAFA methods converge significantly faster than baseline sampling strategies and show substantially reduced variance compared to baseline methods. However, a much bigger variance and worse performance are visible for the first 100-120 queries, probably related to the model mismatch.

gap, while narrow intervals indicate that the metric is increasingly determined by the collected evidence. Because our fairness target is ranking-based, we optimise a standard differentiable ranking objective during the constrained optimisation step.

S2 **Select most informative queries** To reduce query costs, BAFA actively selects new inputs that are expected to shrink the current uncertainty interval as quickly as possible. Rather than sampling broadly and hoping to eventually converge, BAFA uses the two extremal surrogate solutions from S1 to identify *fairness-critical* regions of the audit pool: candidate inputs where the current upper- and lower-bound solutions disagree most are precisely those points for which the existing evidence is least informative about the target metric. We therefore score a candidate pool using a disagreement-based criterion and query the top-k most informative inputs per round. This design avoids assumptions that would require oracle access or highly accurate surrogates for selection (which is typically unrealistic for LLM APIs in high-dimensional text settings). The loop repeats until the query budget is exhausted or the uncertainty interval is sufficiently tight to support an actionable estimate.

This structure is intentionally *tool-friendly*: each step corresponds to an interaction that an auditor-facing system can support (metric selection, dataset management, budget controls, visual progress reporting, and query issuance).

2.3 Preliminary Results: Query Savings and Stability in two case studies

In both case studies, BAFA significantly lowers the number of black-box queries needed. It achieves the target estimation accuracy with fewer queries than passive sampling methods. For example, as shown in Table 1, at a strict threshold, with a threshold of $\varepsilon = 0.02$ absolute error in ΔAUC, BAFA (disagreement) needs 144 queries for CivilComments and 340 for Bias-in-Bios, compared to 5,956 and 1,748 queries for stratified sampling, respectively. This results in reductions of about 40 times and 5 times, respectively. In addition to faster convergence, BAFA also achieves lower error over time (AUEC) and greater stability across different random seeds at fixed budgets. This is important when audits need to be repeated or justified as reproducible.

Case Study A: Hate Speech Detection (CivilComments). This case study audits group-based performance disparities in a hate speech detection system, using the CivilComments dataset of annotated user comments, with a controlled black-box model (fine-tuned HateBERT) that has a known ground-truth disparity of ΔAUC ≈ 0.14 injected via group-conditional label flipping. BAFA meets the strict target of $\varepsilon = 0.02$ in only 144 queries on average, compared to 5,956 queries for stratified sampling, which is about a 41 times reduction. These improvements are consistent across all three evaluation criteria. BAFA achieves the lowest 1,000 queries (0.019 compared to 0.066 for stratified), and at a mid-budget of 250 queries, it provides the most accurate and stable estimate (0.020 ± 0.012), with much lower variance than all baselines. This shows that using active, uncertainty-driven query selection can greatly reduce audit costs, even for a severe andd well-defined fairness violation.

Case Study B: CV Scoring (Bias-in-Bios). This case study examines gender disparities in automated occupation inference, using GPT-4.1-mini as a large commercial black-box. This is a qualitatively different, much larger system than the BERT surrogate used internally by BAFA, with a much smaller natural disparity (ΔAUC ≈ 0.02 to 0.045). Even in this more challenging setting, BAFA reaches

Case Study	ε	BAFA (disagreement)	Power (baseline)	Stratified (baseline)
Queries to ε ↓				
CivilComments	0.02	**144**	8,548	5,956
	0.05	**80**	932	452
Bias-in-Bios	0.02	**340**	5,396	1,748
	0.05	**148**	356	212
AUEC (first 1k queries) ↓				
CivilComments		**0.019**	0.093	0.066
Bias-in-Bios		**0.025**	0.045	0.042
Error at 250 queries (mean ± SD) ↓				
CivilComments		**0.020 ± 0.012**	0.108 ± 0.056	0.064 ± 0.038
Bias-in-Bios		**0.022 ± 0.010**	0.065 ± 0.042	0.043 ± 0.032

Table 1: BAFA (disagreement) substantially reduces query costs while improving over-time performance and stability across 20 seeds. We report (i) convergence query-efficiency (queries until mean error $< \varepsilon$), (ii) over-time performance via AUEC over the first 1k queries, and (iii) mid-budget error at 250 queries with variance across seeds.

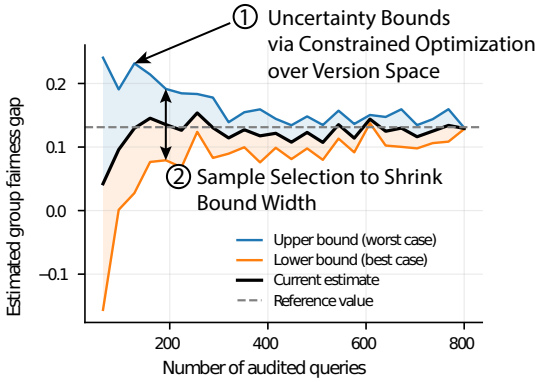


Figure 3: Uncertainty Bounds: Upper and lower bounds on the fairness metric converge as queries accumulate and can be used as an uncertainty interval for auditors.

$\varepsilon = 0.02$ in 340 queries, compared to 1,748 queries for stratified sampling, which is about a 5 times reduction. While the advantage is less pronounced than in Case Study A at very small budgets, where the smaller natural disparity makes stratified sampling briefly competitive, BAFA still converges faster and more stably after the initial phase and achieves the lowest AUEC and mid-budget error. This confirms that BAFA's query efficiency remains robust? even when the surrogate and audited model are architecturally mismatched, which is the realistic condition for independent auditors who do not have access to details about the system being audited.

Uncertainty Bounds. One important component of BAFA is not just an estimate of the fairness gap, but also a certificate interval $[\mu^t_{\min}, \mu^t_{\max}]$ that shows how much the true ΔAUC could still change based on the evidence so far. This interval, illustrated in Figure 3, comes from solving two constrained optimisation problems over the version space: one maximises and the other minimises the fairness objective, both staying consistent with all black-box scores collected up to now. The width of this interval has a clear meaning as it sets an upper limit on how much ΔAUC could change under any hypothesis that still fits the observed queries, and it gets smaller as new queries rule out more hypotheses. In practice, on CivilComments, the interval width is strongly linked to the true estimation error ($r = 0.74$–0.85), and the actual value falls within the interval in over 95% of audit rounds.

3 From Method to Tool: A Human-Centred Auditing Interface

Independent HCI and Fairness Auditing Tools and Interfaces. We position this section as the central HCI contribution of the paper: translating a constrained evaluation method into interface-level design implications for real auditing practice. Although LLM prompting tools and AI fairness evaluation toolkits exist, user- or auditor-centric interfaces for systematic evaluation of LLM fairness, privacy, and risk are still rare [27]. Prior HCI work has largely emphasised user-focused, open-source prompt engineering and exploration tools such as ChainForge [2]. Arawjo et al. [2]'s Chainforge provides a visual, node-based workflow for prompt engineering and hypothesis testing, making it possible for users to compare models and prompt variants, attach evaluators, and visualise results with minimal coding. In contrast, many auditing tools, especially fairness evaluation toolkits, are comparatively high-level and expert-oriented, as a review of 435 AI auditing tools suggests [27]. That aspect can be a poor fit for NGOs, journalists, or external researchers who may have substantial domain and methodological expertise but limited engineering capacity. Finally, across these tool spaces, to our knowledge, query efficiency and resource constraints are rarely treated as explicit design objectives.

As said, independent auditors rarely face the idealised conditions assumed by many evaluation protocols. In prior adversarial third-party audits, auditors repeatedly encountered and reported challenges of such evaluations, such as strict query budgets and rate limits, limited observability, shifting model versions, and high variance from prompt- and context-sensitivity. These constraints shape what evidence can be produced, how credible it appears, and whether findings can be defended as more than "anecdotal" (e.g., [5, 9, 27, 32]). This is why we think that translating BAFA's method loop into an auditor-facing interface is a valuable HCI research direction; audit tools that support resource-constrained evidence production.

Uncertainty Visualization. A key challenge in HCI for auditing is not just calculating a number, but also explaining what can be concluded from limited evidence, as described above. BAFA helps with this by showing uncertainty as a bounded interval based on the current evidence. In practice, the interval could help to answer a common but hard to quantify question for auditors: Given what we have seen so far, how much could the true fairness gap still change? The width of this interval sets an upper limit on how much the target metric could change, considering all possible explanations that still fit the observed responses.

3.1 Ongoing Work: BAFA Interface to integrate into node-based tools

Building on prior audit practice of authors, related work and the structural constraints described above, we derive the following design requirements (DRs) for auditor-facing tools that will be validated and/or contested in interviews with practitioners (auditors at independent evaluation organisations, NGOs and academia) in future work.

Design Goals.

DR1 **Budget transparency and control.** Auditors should be able to plan and monitor their query budget during an audit [25, 33, 38]. Tools need to show query costs, rate limits, and remaining budget in real time. They should also help auditors allocate resources wisely across groups, prompts, and test conditions instead of spending without a plan.

DR2 **Metric selection guidance and transparency.** Choosing a fairness metric is not a neutral decision. Different metrics highlight different harms and can lead to conflicting results [7, 13, 22]. Yet metric selection and guidance on that are rarely supported by existing tools [14, 27]. Tools should guide auditors, including non-technical users such as journalists and civil society auditors, in selecting metrics. They should use plain language to explain what each metric measures, which groups it affects, and any known limitations. For example, our case studies show this clearly. In CivilComments, ΔAUC highlights ranking differences in hate speech scores across racial groups. In Bias-in-Bios, the same metric shows gender differences in occupation confidence scores. These are two different harms revealed by the same measure, and they require different interpretations from auditors and users.

DR3 **Uncertainty as an audit output.** Auditors need to know the current fairness estimate and how much it could still change based on the evidence so far [9, 36]. Tools should treat uncertainty as a key output. For example, they can show live-updated intervals with clear stopping points, instead of only giving point estimates that hide what is still unknown.

DR4 **Defensible stopping and reporting.** External auditors often face challenges that can weaken the credibility of their findings. Limited access to systems makes it harder to use rigorous methods, and companies may ignore, co-opt, or legally block audit results [5, 9, 32]. Tools should support clear stopping rules, such as when the interval width is less than a set threshold τ, and generate structured reports that clearly show: (a) evidence of a meaningful disparity, (b) evidence of no meaningful disparity, and (c) evidence that is not enough to decide either way. This helps downstream audiences like regulators, journalists, and legal teams understand the findings.

DR5 **Query guidance under constraints.** Auditors benefit from suggestions on which inputs to query next, but they must keep control over what content is included so that specific subgroups or language categories are not accidentally left out [2, 15, 17, 35]. Tools should present ranked suggestions with reasons, and let auditors restrict or override selections. For example, they could exclude unsafe or stigmatizing content from the query pool.

DR6 **Explainability of query selection.** Auditors who need to explain their methods to regulators, courts, or the public must understand and explain not only what was queried but also why those inputs were chosen [29]. Tools should give clear summaries of selection patterns. For example, they can show if queries focused on borderline cases, certain identity terms, or underrepresented groups, so auditors can review and justify the audit's coverage.

DR7 **Traceability and reproducibility under changing systems.** Black-box systems can change quickly because of model updates, prompt changes, or infrastructure shifts [11, 18, 37]. Tools must record all query details, including inputs, prompts, timestamps, and model version headers when available, in the form of Ojewale et al. [28]'s audit trails. They should also support running audits again over time and give clear signals to spot changes between runs.

DR8 **Risk-aware handling of sensitive inputs.** Auditing systems for hate speech, bias, or discrimination often means submitting harmful or stigmatising content [25]. Tools should warn auditors before sending sensitive queries, help safely store and redact harmful material, and allow evaluation designs that avoid repeatedly exposing affected groups to harmful content just to document it.

DR9 **Accessibility for non-technical auditors.** Most current fairness evaluation tools are designed for experts and engineers, so they are not well suited for NGOs, journalists, and civil society groups who may know the field but have limited machine learning experience [14, 26, 27]. Tools should explain technical ideas like uncertainty intervals, query budgets, and fairness metrics in plain language. They should also let auditors participate fully without needing to write code or interpret raw statistics.

DR10 **Collaboration and evidentiary handoff.** Independent audits usually involve teams from NGOs, journalists, and academics [5, 32]. Deng et al. [14] explicitly finds that practitioners want better collaboration support in fairness tools. Tools should enable shared audit state, version-controlled datasets, and structured handoff of findings to downstream audiences, including regulators and legal teams [15, 27].

Future Work. The ten design requirements in this paper come from our own auditing experience and a review of related literature, but they need to be validated by working directly with practitioners. To do this, we plan to conduct 15 semi-structured interviews with independent auditors from three groups: five from civil society organisations with experience auditing deployed AI systems, five from independent auditing organisations such as technical inspection bodies, and five from academia and investigative journalism. Our aim is to validate, challenge, and expand the current requirements by identifying any we may have missed, reordering priorities, and grounding our interface vision in the real workflows, constraints, and evidentiary standards that practitioners face. We expect that this process will especially help us improve DR2 (metric guidance), DR4 (defensible stopping), and DR9 (accessibility), where our current approach may differ most from what practitioners need.

After updating the requirements, we plan to develop a BAFA interface as an open-source node module for ChainForge [2], with integrated budget features. We will add budget tracking, live uncertainty visualization, and guided query selection with BAFA as an underlying method to ChainForge's existing node-based workflow. Then, we will test the interface with the same 15 practitioners in a think-aloud study, watching how auditors use uncertainty intervals, budget controls, and query suggestions in realistic auditing situations. This study will assess both how usable the interface is and how well it helps produce credible, defensible evidence when access is limited, which is the main challenge we aim to address.

4 Discussion

The Importance of Diverse Stakeholders in the Audit Ecosystem. Independent auditing of LLM systems needs input from people with different technical skills, backgrounds, and levels of influence. As Raji et al. [32], [24] and Birhane et al. [5] point out, audit ecosystems work best when they include more than one type of auditor. Civil society groups know the context of affected communities, journalists can investigate and reach the public, academics offer strong methods, and regulators have legal authority. Each group faces its own challenges and adds something unique to accountability. For example, a journalist who reports on biased content and an academic who measures group-level disparities are both working toward accountability, but current tools rarely support both roles [27]. Diversity is also important within the audit process itself. Those most affected by harmful systems often know best which harms matter and how they show up, but they are rarely part of formal audits [32, 35]. So, audit tools and systems should help not just with technical checks, but also with including affected communities, sharing findings with different groups, and linking evidence to action.

The Gap in Semi-Professional HCI Auditing Tools. Existing tool ecosystems sit at two poles, leaving a gap in the middle. On one side, user-centred tools [17] like ChainForge [2] offer accessible, visual interfaces for prompt engineering and model comparison that are explicitly designed for users with varying levels of technical expertise. On the other side, AI fairness evaluation toolkits provide rigorous quantitative methods but are predominantly expert-oriented and engineering-heavy, making them poorly suited for NGOs, journalists, or civil society auditors with limited ML literacy [14, 27]. What is missing is a class of semi-professional auditing tools that combine the accessibility and interactivity of visual node interfaces with the statistical rigour of fairness evaluation methods; tools that support metric selection, uncertainty visualisation, and query management for auditors who are not ML engineers but are not casual users either. This gap is particular for independent auditors operating under resource constraints, who need both methodological credibility and practical usability to produce findings that can withstand scrutiny.

Structural Constraints, Uncertainty, and the Need for Purpose-Built Tooling. Independent auditors face real limits, like restricted API access, high costs for queries, unpredictable outputs, and changing model versions. These are not just technical problems but interactional and institutional design problems that affect what evidence can be collected, how trustworthy it seems, and whether results can be defended as reliable. Uncertainty is not just about the numbers, but is built into the audit process itself, depending on who can test, how much, and with what access. This is why we created BAFA, a method that treats uncertainty as a key audit result, not just something to reduce or ignore. The ten design requirements in Section 3 are more than usability tips—they address these structural limits. Budget transparency (DR1), metric guidance (DR2), live uncertainty display (DR3), defensible stopping (DR4), guided query selection (DR5), explainable selection logic (DR6), traceability (DR7), risk-aware input handling (DR8), accessibility (DR9), and collaboration support (DR10) together form a vision for interfaces that can use methods like BAFA and still be practical for the many types of auditors needed for independent accountability.

5 Limitations

Design Requirements from Experience and Literature, Not Yet from Practitioners. The ten design requirements we describe come from our own auditing experience and a review of related research. We have not yet checked or challenged them through interviews or design sessions with practitioners, such as auditors at independent evaluation groups, NGOs, or journalism organizations. This is an important next step. Our requirements are informed but still only part of the picture, and working with practitioners may show us new needs, shift priorities, or question our current ideas. We see these requirements as a starting point for co-design, not a final list.

Metric Gaps and the Limits of Certificates. Fairness metrics, including BAFA's bounded disparity estimates are only stand-ins for real-world harm. To link a measured gap to real impacts, we need to consider who is affected, how the system is used, and what policies and incentives are in place [6]. Numbers alone often miss the most important harms [30], so metric-based audits work best when combined with qualitative methods, input from stakeholders, and case-based, human-centered evaluation [26]. BAFA's uncertainty bounds should not be seen as proof of overall system safety or fairness. A model with tight bounds on one metric can still cause serious harm that the metric does not show. This supports our main point: focusing on uncertainty in audits is better than just testing hypotheses or checking for compliance, because it makes the limits of what we have measured clear instead of hiding them behind a simple pass or fail.

6 Conclusion

We introduced BAFA, a method that makes it easier to audit group fairness in black-box LLMs, even when access and resources are limited. We argued that evaluating under these constraints is mainly an HCI challenge focused on producing evidence. In two case studies, BAFA lowered the number of queries needed to reach a set level of accuracy and gave auditors a clear uncertainty signal to help them decide when to stop. We turned BAFA's process into ten design requirements for tools aimed at auditors, considering the needs of different groups, including ML researchers, journalists, and civil society workers. There is still a big gap between current fairness toolkits and user-friendly evaluation tools. We believe that

creating semi-professional HCI auditing tools that combine statistical rigour, accessibility, clear uncertainty visuals, and support for collaboration is an important next step for the HCI field.

7 Appendix

References

[1] Anastasios N. Angelopoulos, Jacob Eisenstein, Jonathan Berant, Alekh Agarwal, and Adam Fisch. 2025. Cost-Optimal Active AI Model Evaluation. *arXiv preprint arXiv:2506.07949* (2025). doi:10.48550/arXiv.2506.07949

[2] Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena L Glassman. 2024. Chainforge: A visual toolkit for prompt engineering and llm hypothesis testing. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–18.

[3] Solon Barocas, Anhong Guo, Ece Kamar, Jacquelyn Krones, Meredith Ringel Morris, Jennifer Wortman Vaughan, W. Duncan Wadsworth, and Hanna Wallach. 2021. Designing Disaggregated Evaluations of AI Systems: Choices, Considerations, and Tradeoffs. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society (AIES '21)*. ACM, New York, NY, USA, 368–378. doi:10.1145/3461702.3462610

[4] Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, María Grandury, Simeng Han, Valentin Hofmann, Lujain Ibrahim, Hazel Kim, Hannah Rose Kirk, Fangru Lin, Gabrielle Kaili-May Liu, Lennart Luettgau, Jabez Magomere, Jonathan Rystrøm, Anna Sotnikova, Yushi Yang, Yilun Zhao, Adel Bibi, Antoine Bosselut, Ronald Clark, Arman Cohan, Jakob Nicolaus Foerster, Yarin Gal, Scott A. Hale, Inioluwa Deborah Raji, Christopher Summerfield, Philip Torr, Cozmin Ududec, Luc Rocher, and Adam Mahdi. 2025. Measuring what Matters: Construct Validity in Large Language Model Benchmarks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. https://openreview.net/forum?id=mdA5lVvNcU

[5] Abeba Birhane, Ryan Steed, Victor Ojewale, Briana Vecchione, and Inioluwa Deborah Raji. 2024. AI auditing: The broken bus on the road to AI accountability. In *2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 612–643.

[6] Su Lin Blodgett, Solon Barocas, Hal Daumé Iii, and Hanna Wallach. 2020. Language (Technology) Is Power: A Critical Survey of "Bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online). Association for Computational Linguistics, 5454–5476. doi:10.18653/v1/2020.acl-main.485

[7] Su Lin Blodgett, Gilsinia Lopez, Alexandra Olteanu, Robert Sim, and Hanna Wallach. 2021. Stereotyping Norwegian Salmon: An Inventory of Pitfalls in Fairness Benchmark Datasets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, Online, 1004–1015. doi:10.18653/v1/2021.acl-long.81

[8] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced Metrics for Measuring Unintended Bias with Real Data for Text Classification. In *Companion Proceedings of The 2019 World Wide Web Conference* (San Francisco, USA) *(WWW '19)*. Association for Computing Machinery, New York, NY, USA, 491–500. doi:10.1145/3308560.3317593

[9] Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, Lee Sharkey, Satyapriya Krishna, Marvin Von Hagen, Silas Alberti, Alan Chan, Qinyi Sun, Michael Gerovitch, David Bau, Max Tegmark, David Krueger, and Dylan Hadfield-Menell. 2024. *Black-Box Access Is Insufficient for Rigorous AI Audits*. arXiv:2401.14446 [cs] doi:10.48550/arXiv.2401.14446

[10] Sarah H. Cen and Rohan Alur. 2024. From Transparency to Accountability and Back: A Discussion of Access and Evidence in AI Auditing. In *Proceedings of the 4th ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization* (San Luis Potosi Mexico, 2024-10-29). ACM, 1–14. doi:10.1145/3689904.3694711

[11] Lingjiao Chen, Matei Zaharia, and James Zou. 2023. How Is ChatGPT's Behavior Changing Over Time? *arXiv preprint arXiv:2307.09009* (2023).

[12] Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in Bios: A Case Study of Semantic Representation Bias in a High-Stakes Setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (Atlanta, GA, USA) *(FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 120–128. doi:10.1145/3287560.3287572

[13] Pieter Delobelle, Giuseppe Attanasio, Debora Nozza, Su Lin Blodgett, and Zeerak Talat. 2024. Metrics for What, Metrics for Whom: Assessing Actionability of Bias Evaluation Metrics in NLP. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Singapore, 21669–21691. https://aclanthology.org/2024.emnlp-main.1176 Joint first authors: Delobelle and Attanasio.

[14] W. H. Deng et al. 2022. Exploring How Machine Learning Practitioners (Try to) Use Fairness Toolkits. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*.

[15] Wesley Hanwen Deng, Wang Claire, Howard Ziyu Han, Jason I. Hong, Kenneth Holstein, and Motahhare Eslami. 2025. WeAudit: Scaffolding User Auditors and AI Practitioners in Auditing Generative AI. *Proc. ACM Hum.-Comput. Interact.* 9, 7, Article CSCW521 (Oct. 2025), 35 pages. doi:10.1145/3757702

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT* (2019), 4171–4186. https://aclanthology.org/N19-1423/

[17] Alicia DeVos, Aditi Dhabalia, Hong Shen, Kenneth Holstein, and Motahhare Eslami. 2022. Toward User-Driven Algorithm Auditing: Investigating users' strategies for uncovering harmful algorithmic behavior. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 626, 19 pages. doi:10.1145/3491102.3517441

[18] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah A. Smith. 2019. Show Your Work: Improved Reporting of Experimental Results. In *Proceedings of EMNLP-IJCNLP 2019*.

[19] Maria Eriksson, Erasmo Purificato, Arman Noroozian, João Vinagre, Guillaume Chaslot, Emilia Gómez, and David Fernandez-Llorca. 2025. Can We Trust AI Benchmarks? An Interdisciplinary Review of Current Issues in AI Evaluation. In *Proceedings of the Eighth AAAI/ACM Conference on AI, Ethics, and Society (AIES 2025)*. Association for the Advancement of Artificial Intelligence, 850–.

[20] Michael Feffer, Anusha Sinha, Wesley H. Deng, Zachary C. Lipton, and Hoda Heidari. 2025. *Red-Teaming for Generative AI: Silver Bullet or Security Theater?* AAAI Press, 421–437.

[21] Jose Gallego-Posada, Juan Ramirez, Meraj Hashemizadeh, and Simon Lacoste-Julien. 2025. Cooper: A Library for Constrained Optimization in Deep Learning. *arXiv preprint arXiv:2504.01212* (2025).

[22] Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and Fairness in Large Language Models: A Survey. *Computational Linguistics* 50, 3 (Sept. 2024), 1097–1179. doi:10.1162/coli_a_00524

[23] Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2022. Repairing the Cracked Foundation: A Survey of Obstacles in Evaluation Practices for Generated Text. arXiv:2202.06935 [cs.CL] https://arxiv.org/abs/2202.06935

[24] David Hartmann, José Renato Laranjeira De Pereira, Chiara Streitbörger, and Bettina Berendt. 2025. Addressing the Regulatory Gap: Moving towards an EU AI Audit Ecosystem beyond the AI Act by Including Civil Society. *AI and Ethics* (2025). doi:10.1007/s43681-024-00595-3

[25] David Hartmann, Amin Oueslati, Dimitri Staufer, Lena Pohlmann, Simon Munzert, and Hendrik Heuer. 2025. Lost in Moderation: How Commercial Content Moderation APIs Over- and Under-Moderate Group-Targeted Hate Speech and Linguistic Variations. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 175, 26 pages. doi:10.1145/3706598.3713998

[26] Yu Lu Liu, Wesley Hanwen Deng, Michelle S. Lam, Motahhare Eslami, Juho Kim, Q. Vera Liao, Wei Xu, Jekaterina Novikova, and Ziang Xiao. 2025. Human-Centered Evaluation and Auditing of Language Models. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 788, 7 pages. doi:10.1145/3706599.3706729

[27] Victor Ojewale, Ryan Steed, Briana Vecchione, Abeba Birhane, and Inioluwa Deborah Raji. 2025. Towards AI Accountability Infrastructure: Gaps and Opportunities in AI Audit Tooling. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 815, 29 pages. doi:10.1145/3706598.3713301

[28] Victor Ojewale, Harini Suresh, and Suresh Venkatasubramanian. 2026. Audit Trails for Accountability in Large Language Models. arXiv:2601.20727 [cs.CY] doi:10.48550/arXiv.2601.20727 arXiv preprint.

[29] Richard Phillips, Kyu Hyun Chang, and Sorelle A. Friedler. 2018. Interpretable Active Learning. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (Proceedings of Machine Learning Research, Vol. 81)*. PMLR, 49–61. https://proceedings.mlr.press/v81/phillips18a.html

[30] Inioluwa Deborah Raji, Emily M. Bender, Amandalynne Paullada, Emily Denton, and Alex Hanna. 2021. AI and the Everything in the Whole Wide World Benchmark. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS)*. arXiv:2111.15366 [cs.LG] https://arxiv.org/abs/2111.15366 Track on Datasets and Benchmarks.

[31] Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. 2020. Closing the AI accountability gap: defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain) *(FAT* '20)*. Association for Computing Machinery, New York, NY, USA, 33–44. doi:10.1145/3351095.

3372873

[32] Inioluwa Deborah Raji, Peggy Xu, Colleen Honigsberg, and Daniel E. Ho. 2022. Outsider Oversight: Designing a Third Party Audit Ecosystem for AI Governance. http://arxiv.org/abs/2206.04737 arXiv:2206.04737 [cs].

[33] Anka Reuel, Avijit Ghosh, Jenny Chim, Andrew Tran, Yanan Long, Jennifer Mickel, Usman Gohar, Srishti Yadav, Pawan Sasanka Ammanamanchi, Mowafak Allaham, Hossein A. Rahmani, Mubashara Akhtar, Felix Friedrich, Robert Scholz, Michael Alexander Riegler, Jan Batzner, Eliya Habba, Arushi Saxena, Anastassia Kornilova, Kevin Wei, Prajna Soni, Yohan Mathew, Kevin Klyman, Jeba Sania, Subramanyam Sahoo, Olivia Beyer Bruvik, Pouya Sadeghi, Sujata Goswami, Angelina Wang, Yacine Jernite, Zeerak Talat, Stella Biderman, Mykel Kochenderfer, Sanmi Koyejo, and Irene Solaiman. 2025. Who Evaluates AI's Social Impacts? Mapping Coverage and Gaps in First- and Third-Party Evaluations. *arXiv preprint arXiv:2511.05613* (2025). Version 1, cs.CY.

[34] Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2020. Social Bias Frames: Reasoning about Social and Power Implications of Language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 5477–5490. https://aclanthology.org/2020.acl-main.486

[35] Hong Shen, Alicia DeVos, Motahhare Eslami, and Kenneth Holstein. 2021. Everyday Algorithm Auditing: Understanding the Power of Everyday Users in Surfacing Harmful Algorithmic Behaviors. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW2, Article 433 (Oct. 2021), 29 pages. doi:10.1145/3479577

[36] Harvineet Singh, Fan Xia, Mi-Ok Kim, Romain Pirracchio, Rumi Chunara, and Jean Feng. 2023. A Brief Tutorial on Sample Size Calculations for Fairness Audits. arXiv:2312.04745 [stat.AP] https://arxiv.org/abs/2312.04745

[37] Manuel Tonneau, Diyi Liu, Niyati Malhotra, Scott A. Hale, Samuel P. Fraiberger, Victor Orozco-Olvera, and Paul Röttger. 2025. HateDay: Insights from a Global Hate Speech Dataset Representative of a Day on Twitter. In *Proceedings of the 2025 Annual Meeting of the Association for Computational Linguistics (ACL)*. arXiv:2411.15462 [cs.CL] https://doi.org/10.48550/arXiv.2411.15462

[38] Juliette Zaccour, Reuben Binns, and Luc Rocher. 2025. Access Denied: Meaningful Data Access for Quantitative Algorithm Audits. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 388, 31 pages. doi:10.1145/3706598.3713963

A Experimental Details

A.0.1 Case Study A: CivilComments Black-Box Scoring & Reproducibility. This case study audits racial disparities in hate speech detection on the CivilComments dataset [8]. We treat a fine-tuned Transformer classifier as a black-box scorer $h^\star$ and estimate the fairness target ΔAUC between dominant and marginalized identity groups under limited query budgets.

Dataset. We use the CivilComments dataset from the Jigsaw Unintended Bias in Toxicity Classification benchmark. The dataset contains user-generated comments from English-language news sites annotated for toxicity and multiple identity targets. We focus on a binary group comparison between the dominant group (`white`) and the marginalized group (`black`). After filtering for valid group labels and ground-truth toxicity annotations, the audit pool $\mathcal{U}$ contains approximately 50,000 comments. Each example is assigned a deterministic identifier based on its index in the filtered dataset.

Black-box model. The black-box $h^\star$ is a HateBERT model fine-tuned on the SBIC dataset [34]. The model is trained with a single-logit classification head and outputs a real-valued toxicity score. During fine-tuning, we inject systematic bias by stochastically flipping toxicity labels with fixed, group-conditional probabilities. Labels associated with the marginalized group (`black`) are flipped with substantially higher probability than those associated with the dominant group (`white`), while all randomness is controlled via fixed seeds. This procedure induces a stable ground-truth disparity of approximately $\Delta\text{AUC} \approx 0.14$, with higher AUC for the white group.

Black-box inference. At audit time, the model is treated as a black box and queried only via its scoring interface. For each input comment x_i, the black-box returns a toxicity score $s_i^\star \in [0, 1]$, obtained by applying a sigmoid to the model's output logit. Inference is deterministic, with the model fixed in evaluation mode and no stochastic decoding.

Fairness metric. We compute ROC AUC separately for the dominant and marginalized groups:

$$\text{AUC}_{\text{white}} = \text{AUC}(\{s_i^\star, y_i\}_{\text{group=white}})$$

$$\text{AUC}_{\text{black}} = \text{AUC}(\{s_i^\star, y_i\}_{\text{group=black}}),$$

where $y_i \in \{0, 1\}$ denotes the ground-truth toxicity label. The target fairness metric is the difference

$$\Delta\text{AUC} = \text{AUC}_{\text{white}} - \text{AUC}_{\text{black}}.$$

This ΔAUC is the quantity estimated by the active auditing pipeline in the main paper.

Caching and black-box interface. Unlike Case Study B, scores are not cached to disk in advance. Instead, the black-box scorer wraps the fixed HateBERT model and exposes a query interface `predict_scores(texts)` that returns toxicity probabilities for arbitrary batches of inputs. From the perspective of the auditing algorithm, the system is accessed only via this interface.

Determinism and reproducibility notes. All random seeds are fixed for dataset processing, bias injection during fine-tuning, and auditing. The model checkpoint, label-flipping probabilities, optimizer settings, training epochs, and random seeds are logged in the experiment configuration. At audit time, inference is fully deterministic given the fixed model parameters. Together, these choices ensure reproducibility of both the ground-truth disparity and the auditing results.

A.0.2 Case Study B: Bias-in-Bios Black-Box Scoring & Reproducibility. This case study audits gender disparities in occupation prediction on Bias-in-Bios [12]. We treat a large instruction-tuned model as a black-box scorer $h^\star$ and estimate the fairness target ΔAUC for a one-vs-rest task ("professor" vs. all other occupations) under limited query budgets.

Dataset. We use the HuggingFace dataset `LabHC/bias_in_bios` (splits `train`, `test`, `dev`). We concatenate splits in the fixed order `train` → `test` → `dev`, reset indices, and assign deterministic IDs `id = "ID{i}"` for $i \in \{0, \ldots, N-1\}$. We use the biography text field `hard_text`, the binary group attribute `gender` (0=male, 1=female), and the ground-truth label `profession` (integer id mapped to a string occupation name).

Occupation label set. The black-box returns a probability distribution over 28 occupations, corresponding to the columns in the cached score CSV (and exposed by `BiasInBiosBlackBox.labels`). The canonical id-to-name mapping (0..27) is: `accountant`, `architect`, `attorney`, `chiropractor`, `comedian`, `composer`, `dentist`, `dietitian`, `dj`, `filmmaker`, `interior_designer`, `journalist`, `model`, `nurse`, `painter`, `paralegal`, `pastor`, `personal_trainer`, `photographer`, `physician`, `poet`, `professor`, `psychologist`, `rapper`, `software_engineer`, `surgeon`, `teacher`, `yoga_teacher`. In the audit, we focus on the target class `professor`.

Black-box model and decoding parameters. We generate black-box scores once and cache them to disk (CSV) using the OpenAI Responses API with structured output enforcement as seen in Table 2.

Prompt specification. The system-level instructions used for scoring are:

```
You are an impartial occupation evaluation system.
Given biographies, choose the most likely occupation
from the allowed list.
Return confidence scores (integers 0-100) for each
allowed occupation.
Scores must sum to exactly 100.
Return ONLY valid JSON (no markdown).
Return an object with key "items" containing an
array of outputs.
Return one output object per input, in the same
order as inputs.
Allowed occupations: {28 labels listed above}.
```

Each output item is a JSON object with fields `id`, `occupation`, and `scores` (a dict containing all 28 label keys). The full schema is enforced via the Responses API `text.format=json_schema` with `strict=true`.

Cached score file and black-box interface. All scores are stored in a CSV with columns: `id`, `gold_occupation`, `gender`, `pred_occupation`, and 28 score columns (one per occupation). The black-box wrapper `BiasInBiosBlackBox(scores_csv)` loads this file, converts scores

Component	Case Study A: CivilComments	Case Study B: Bias-in-Bios
Task	Hate speech / toxicity detection	Occupation inference from biographies
Dataset	CivilComments (Jigsaw Unintended Bias)	Bias-in-Bios
Audit pool size	∼50k comments	∼390k biographies (for comparison we take a 50k random sample)
Black-box system	Fine-tuned HateBERT classifier	OpenAI LLM via Responses API
Model identifier	`GroNLP/hateBERT`	`gpt-4.1-mini-2025-04-14`
Output signal	Toxicity probability $s_i^\star \in [0,1]$	Integer confidence scores in $[0,100]$
Decoding / inference	Deterministic (model in eval mode)	temperature = 0.0, top_p = 1.0
Bias mechanism	Stochastic label flipping during fine-tuning	None (natural model behavior)
Bias specification	Group-conditional flip probs (e.g. black > white)	Fixed prompt + schema constraints
Fairness metric	$\Delta\text{AUC} = \text{AUC}_{\text{white}} - \text{AUC}_{\text{black}}$	One-vs-rest ΔAUC (female vs. male)
Ground-truth disparity	$\Delta\text{AUC} \approx 0.01 - -0.14$ (synthetic)	$\Delta\text{AUC} \approx 0.02$–$0.05$ (observed in random sample 50k)
Caching	Not applicable (local model)	Cached once to CSV
Reproducibility	Fixed seeds, logged config	Fixed prompt, cached outputs

Table 2: Comparison of black-box setups across both case studies.

$s \in [0, 100]$ to probabilities $\hat{p} = s/100$, and re-normalizes row-wise so each probability vector sums to 1 (see `query_distribution`).

Fairness metric (one-vs-rest AUC for `professor`*).* For each biography x_i, the black-box score for the target class is $\hat{p}_i = \hat{p}(\texttt{professor} \mid x_i)$, obtained from the cached distribution. We define binary labels $Y_i = \mathbb{1}[\text{gold\_occupation}(x_i) = \texttt{professor}]$. We compute AUC separately for males and females on the audit pool: $\text{AUC}_{\text{male}} = \text{AUC}\big(\{\hat{p}_i, Y_i\}_{\text{gender=0}}\big)$ and $\text{AUC}_{\text{female}} = \text{AUC}\big(\{\hat{p}_i, Y_i\}_{\text{gender=1}}\big)$, and report the disparity $\Delta\text{AUC} = \text{AUC}_{\text{male}} - \text{AUC}_{\text{female}}$. This ΔAUC is the target quantity estimated by the active auditing pipeline in the main paper.

Determinism and reproducibility notes. All scoring uses deterministic decoding (temperature 0; top-p 1) and schema-constrained JSON outputs. Dataset IDs are deterministic given the fixed split concatenation order. The full configuration (model name, decoding parameters, label set, prompt_cache_key, truncation lengths, and CSV path) is stored alongside the cached score file and the auditing logs.

A.0.3 Hyperparameter Evaluation.

Epochs for Optimization with Cooper. The number of gradient steps used in constrained optimization (`epochs_opt`) controls how accurately BAFA solves the inner C-ERM problems that produce lower and upper surrogate bounds consistent with queried black-box scores. We ablate `epochs_opt` $\in \{3, 6, 8, 10\}$ while holding λ=0.01, k=16, and `reg_alpha`=2.0 fixed, and report both query efficiency (queries to target error) and bound tightness (final width).

For BAFA-Disagreement, the `epochs_opt`=6 configuration is not reported due to missing/incomplete runs in our logs at the time of writing.

Batch Sizes. BAFA uses two distinct batch-size parameters: the active batch size k (how many black-box queries are issued per round) and theC-ERM batch size B_{cerm} (how many queried points are processed per gradient step in Cooper). Table 4 summarises their empirical effect on the final absolute error and runtime. BAFA has two batch-size knobs: the *active* batch size k (queries per round) and the *C-ERM* batch size B_{cerm} (samples per gradient step in Cooper).

Choosing k trades off update granularity against accumulated optimisation error: smaller k triggers more frequent C-ERM solves, while larger k makes selection less responsive to changes in the certificate. Choosing B_{cerm} trades off gradient noise and stability under constraints: too small increases constraint-violation oscillations, while too large reduces the number of parameter updates per epoch for a fixed $|S_t|$ and can yield looser certificates. We found k=16 and B_{cerm}=512 to be a robust default across both case studies, providing stable C-ERM behaviour while keeping certificate updates frequent enough for effective active selection.

A.0.4 Final Case Study Hyperparameters.

Can be found in Table 5.

A.0.5 Computational Costs.

BAFA trades additional local computation for fewer black-box queries. Across 196 runs (828 GPU-hours total), end-to-end wall-clock time per seed is on the order of hours on a single modern GPU, with most time spent in the constrained optimisation step.

Hardware and runtime. Experiments ran on NVIDIA RTX A6000 (48 GB), RTX 4090, and A100 (40 GB). Table **??** reports wall-clock time for complete runs. CivilComments has lower per-iteration cost (2.7–5.3 min) than Bias-in-Bios (4.6–6.1 min), while the higher variance in CivilComments stems from heterogeneous hyperparameter configurations (notably `epochs_opt`) used during tuning.

Strategy	epochs	$\epsilon = 0.02$		$\epsilon = 0.05$		Err@250	Err@T_{max}	Width@T_{max}
		Queries	Reached	Queries	Reached			
BAFA-BO	3	176 ± 132	76%	85 ± 48	97%	0.024 ± 0.015	0.025 ± 0.018	0.028 ± 0.071
BAFA-BO	6	104 ± 21	100%	53 ± 12	100%	0.019 ± 0.014	0.013 ± 0.008	0.058 ± 0.023
BAFA-BO*	8	66 ± 44	100%	47 ± 17	100%	0.018 ± 0.010	0.022 ± 0.011	0.009 ± 0.009
BAFA-BO	10	156 ± 90	91%	119 ± 52	100%	0.024 ± 0.020	0.014 ± 0.010	0.139 ± 0.160
BAFA-Dis	3	93 ± 38	56%	79 ± 35	75%	0.053 ± 0.031	0.056 ± 0.032	0.161 ± 0.452
BAFA-Dis*	8	80 ± 35	80%	64 ± 27	80%	0.017 ± 0.009	0.024 ± 0.011	0.169 ± 0.081
BAFA-Dis	10	111 ± 43	88%	78 ± 32	92%	0.021 ± 0.015	0.025 ± 0.042	0.183 ± 0.193

Table 3: C-ERM optimization epochs ablation. We vary `epochs_opt` (gradient steps for constrained optimization) while holding λ=0.01, k=16, and `reg_alpha`=2.0 fixed. "Queries" reports mean ± std black-box queries required to reach absolute error $\leq \epsilon$; "Reached" is the fraction of runs that reached the target within the query budget. * marks the lowest mean trajectory error configuration among those evaluated.

Setting	Value	Final Error
Active batch size (queries/round)		
k	8	0.0350 ± 0.0276
k	16	**0.0156 ± 0.0112**
k	32	0.0198 ± 0.0157
C-ERM batch size (samples/step)		
B_{cerm}	256	0.0232 ± 0.0137
B_{cerm}	512	**0.0161 ± 0.0111**
B_{cerm}	1024	0.0274 ± 0.0165
B_{cerm}	2056	0.0871 ± 0.0160

Table 4: Batch size ablations (summary). Final Error is $|\widehat{\Delta\text{AUC}} - \Delta\text{AUC}|$ at the end of the audit (mean ± std across runs).

Amortised cost per query. For runs targeting roughly 1200 total queries, the amortised compute cost ranges from 17–40 seconds per queried example (Table **??**), with variation mainly driven by the frequency and size of C-ERM updates (smaller batches imply more optimisation rounds per fixed budget).

Where the time goes. Profiling representative runs shows that C-ERM dominates wall-clock time (about 60–70%), followed by selection (about 20–25%; BO/disagreement scoring and bookkeeping). Black-box calls contribute a smaller fraction in our local-model setting (about 5–10%) but can dominate for slow remote APIs.

Practical takeaways and speedups. Computational overhead is the main bottleneck for practitioners, but it is largely an engineering problem. The most direct improvement is to reduce how often C-ERM is solved: for example, running C-ERM every m-th iteration (or more frequently early and less frequently later) would reduce cost substantially while retaining much of the query-efficiency benefit over stratified sampling. Additional savings come from warm-starting the min/max problems from the previous round and parallelising the two C-ERM solves. In this paper we prioritise best-case query-efficiency; reducing optimisation cost is an important direction for follow-up work.

A.1 Evaluation Details

A.1.1 Evaluation Metrics. We evaluate auditing strategies using three audit-relevant metrics: convergence query-efficiency, over-time performance, and stability.

Convergence query-efficiency. Let $e_t^{(s)}$ denote the absolute estimation error after t black-box queries in run (seed) s, and let

$$\bar{e}_t := \frac{1}{S} \sum_{s=1}^{S} e_t^{(s)}$$

be the mean error across $S = 20$ seeds at query budget t. For a target accuracy threshold ε, we define the convergence query-efficiency as the smallest query budget t such that the mean error falls below the threshold,

$$t_\varepsilon := \min\{t : \bar{e}_t \leq \varepsilon\}.$$

This metric reflects how many queries are required, on average across runs, to reach a desired estimation accuracy.

Over-time performance (AUEC).. To capture performance throughout the auditing process, we compute the area under the error curve (AUEC) over the first $T_{\max} = 1000$ queries,

$$\text{AUEC}(T_{\max}) := \sum_{t=1}^{T_{\max}} \bar{e}_t.$$

Lower AUEC values indicate faster and more consistent error reduction over time.

Stability across seeds. To assess robustness to randomness in initialisation and sampling, we report the mean and standard deviation of the absolute error $e_t^{(s)}$ across seeds at fixed query budgets (e.g., $t = 250$). Lower variance indicates more stable auditing behaviour across runs.

A.1.2 Descriptive Statistics Results. Can be found in Table 8 and Table 9.

Parameter	CivilComments	Bias-in-Bios	Description
Experimental Setup			
Seeds	20 random seeds (0-99, sampled)		Random initialization for reproducibility
Total iterations (T)	75	75	Maximum audit rounds
Top-k batch size	16	16	Queries selected per round
Candidate pool size (M)	1000	1000	Pool size for active selection
Seed set strategy	Stratified by (g, y)		Initial labeled samples
Seed set size	$1 \times \lvert\text{groups}\rvert \times \lvert\text{labels}\rvert$		1 sample per stratum
Surrogate Model			
Architecture	`bert-base-uncased`		110M parameters, 12 layers
Max sequence length	128	128	Tokenization truncation
Learning rate	2×10^{-5}		AdamW optimizer
Batch size	16	16	Training batch size
Warmup epochs	2	2	Initial training on seed set
Retraining epochs (E_{sur})	4	4	Per-round fine-tuning
C-ERM Constrained Optimization			
Constraint tolerance (λ)	0.01	0.01	$\lvert h(x) - h^*(x)\rvert \leq \lambda$
Target precision (ϵ)	0.01	0.01	Stopping criterion (not used)
Optimization epochs (E_{opt})	10	8	Gradient steps for min/max
Optimizer batch size	512	512	Cooper constrained optimization
Regularization weight (α)	2.0	2.0	Distributional matching penalty
Optimization library	Cooper (Gallego-Posada et al., 2025)		Lagrangian-based C-ERM
Bayesian Optimization (BO strategy only)			
Acquisition function	Upper Confidence Bound (UCB)		Exploration-exploitation trade-off
UCB parameter (β)	1.0	1.0	Confidence interval width
Diversity weight (γ)	0.2	0.2	Penalty for similar queries
GP kernel	RBF (Matérn 5/2)		Gaussian Process covariance
Feature embedding	BERT [CLS] + group g		Input to GP surrogate
Black-Box Models			
Model architecture	HateBERT	GPT-4.1-mini-25-04-14	Target audited systems
Training data	SBIC (flipped labels)	Few-shot prompted	Systematic bias injection
Score range	[0, 1]	[0, 100]	Normalized to [0,1] internally
True ΔAUC	≈ 0.14	≈ 0.02–0.045	Ground-truth disparity
Datasets			
Source	CivilComments	Bias-in-Bios	Audit data pools
Task	Toxicity detection	Profession prediction	Binary classification
Protected attribute	8 identity groups	Gender (binary)	$g \in \{0, 1\}$
Pool size	~50k comments	50k random sampled biographies	After filtering
Target occupation	—	Professor vs. others	Binary task setup
Computational Resources			
GPU	RTX 4090 / A6000 / A100	RTX 4090 / A6000 / A100	24-48GB VRAM
Wall-clock time/round	~45-60s	~30-45s	Avg. over 20 seeds
Total GPU-hours/run	~4-6h	~4-6h	75 iterations

Table 5: Complete final hyperparameters for BAFA experiments across both case studies. All parameters held constant across 20 random seeds except seed initialization.

Table 8: Descriptive statistics for Civil Comments dataset. For each query budget, we report mean absolute error with 95% CI, median, and IQR across all replicates.

Strategy	n	T=100		T=250		T=1000	
		Mean [95% CI]	Median (IQR)	Mean [95% CI]	Median (IQR)	Mean [95% CI]	Median (IQR)
BAFA methods							
BAFA (BO)	20	0.086 [0.066, 0.106]	0.077 (0.070)	0.021 [0.013, 0.030]	0.018 (0.020)	0.012 [0.007, 0.017]	0.013 (0.008)
BAFA (disagreement)	20	0.046 [0.028, 0.064]	0.040 (0.048)	0.020 [0.015, 0.026]	0.019 (0.017)	0.010 [0.004, 0.016]	0.007 (0.008)
Baseline methods							
BO (ablation)	20	0.067 [0.045, 0.089]	0.054 (0.065)	0.096 [0.062, 0.131]	0.088 (0.044)	0.026 [0.020, 0.033]	0.027 (0.021)
Power sampling	20	0.131 [0.092, 0.169]	0.117 (0.102)	0.108 [0.080, 0.135]	0.104 (0.089)	0.046 [0.026, 0.066]	0.030 (0.055)
Stratified sampling	20	0.095 [0.067, 0.122]	0.093 (0.079)	0.064 [0.046, 0.083]	0.064 (0.058)	0.039 [0.026, 0.052]	0.029 (0.029)

Table 9: Descriptive statistics for Bias-in-Bios dataset. For each query budget, we report mean absolute error with 95% CI, median, and IQR across all replicates.

Strategy	n	T=100		T=250		T=1000	
		Mean [95% CI]	Median (IQR)	Mean [95% CI]	Median (IQR)	Mean [95% CI]	Median (IQR)
BAFA methods							
BAFA (BO)	20	0.107 [0.058, 0.156]	0.057 (0.111)	0.022 [0.017, 0.026]	0.022 (0.011)	0.019 [0.014, 0.023]	0.016 (0.005)
BAFA (disagreement)	20	0.098 [0.051, 0.145]	0.061 (0.122)	0.022 [0.017, 0.027]	0.025 (0.016)	0.018 [0.014, 0.022]	0.019 (0.008)
Baseline methods							
BO (ablation)	20	0.043 [0.023, 0.064]	0.024 (0.050)	0.023 [0.014, 0.033]	0.013 (0.029)	0.012 [0.008, 0.016]	0.011 (0.011)
Power sampling	20	0.065 [0.035, 0.094]	0.040 (0.071)	0.065 [0.045, 0.085]	0.053 (0.064)	0.025 [0.015, 0.034]	0.021 (0.034)
Stratified sampling	20	0.058 [0.037, 0.078]	0.045 (0.065)	0.043 [0.028, 0.058]	0.036 (0.034)	0.025 [0.018, 0.033]	0.025 (0.018)