FindTheFlaws: Annotated Errors for Detecting Flawed Reasoning and Scalable Oversight Research

Gabriel Recchia [1*], Chatrik Singh Mangat[2*], Issac Li [3], Gayatri Krishnakumar [4]

[1]Modulo Research
[2]Vector Research
[3]Princeton University
[4]Impact Academy
gabe@moduloresearch.com

Abstract

As AI models tackle increasingly complex problems, ensuring reliable human oversight becomes more challenging due to the difficulty of verifying solutions. Approaches to scaling AI supervision include debate, in which two agents engage in structured dialogue to help a judge evaluate claims; critique, in which models identify potential flaws in proposed solutions; and prover-verifier games, in which a capable 'prover' model generates solutions that must be verifiable by a less capable 'verifier'. Evaluations of the scalability of these and similar approaches to difficult problems benefit from datasets that include (1) long-form expert-verified correct solutions and (2) long-form flawed solutions with annotations highlighting specific errors, but few are available. To address this gap, we present FindTheFlaws, a group of five diverse datasets spanning medicine, mathematics, science, coding, and the Lojban language. Each dataset contains questions and long-form solutions with expert annotations validating their correctness or identifying specific error(s) in the reasoning. We evaluate frontier models' critiquing capabilities and observe a range of performance that can be leveraged for scalable oversight experiments: models performing more poorly on particular datasets can serve as judges/verifiers for more capable models.

Code & Datasets — https://github.com/modulo-research/findtheflaws

Extended Version — https://arxiv.org/abs/2503.22989

1 Introduction

As AI systems become more sophisticated, ensuring reliable human oversight becomes a growing challenge: verifying AI-generated solutions is often difficult, even for domain experts. Limitations of human feedback reduce our ability to trust AI in high-stakes scenarios and raise concerns about robustness, reliability, and alignment (Amodei et al. 2016). Researchers have therefore proposed protocols for 'scalable oversight' (Amodei et al. 2016; Bowman et al. 2022; Irving, Christiano, and Amodei 2018; Christiano, Shlegeris, and Amodei 2018; Brown-Cohen, Irving, and Piliouras 2025): methods that could theoretically allow humans to effectively evaluate AI systems' outputs as AI capabilities increase, even as the task of verifying these outputs becomes too difficult or costly for humans to accomplish directly.

Most scalable oversight protocols ultimately aim to help a judge (a human or trusted model, such as an AI judge trained to predict human judgments) to identify errors in arguments produced by potentially misaligned or deceptive models. However, accurate labels for errors in long-form reasoning are expensive and time-consuming to produce. As a result, existing work has largely focused on whether oversight protocols can help judges arrive at correct binary decisions about solution validity (Bowman et al. 2022; Irving, Christiano, and Amodei 2018; Kenton et al. 2024; Khan et al. 2024; Kirchner et al. 2024), with far less attention paid to the key question of whether judges actually identify the true underlying problems when they reject flawed solutions (but see Lightman et al. (2023); Uesato et al. (2022)). While achieving correct judgments might seem sufficient, we cannot be as confident that these approaches will generalize without understanding the specific *nature* of the errors identified—what it is about the argument that causes it to fail, which can be described by a natural language explanation or an indication of which step of the argument is flawed. In the following paragraphs, we highlight how studies on the efficacy of three families of scalable oversight protocols—critique, debate, and prover-verifier games—could benefit from access to this kind of ground-truth error information. We then present our FindTheFlaws datasets as a resource for conducting such analyses.

Critique models are trained to write natural language assessments that identify potential flaws or limitations in AI-generated outputs, aiming to help human or AI evaluators more effectively assess complex solutions (Saunders et al. 2022). Sun et al. (2024a) enumerate several works that make use of LLM-generated critiques to improve LLM outputs, and propose methods for automated evaluation of critique quality which yield scores that are more correlated with human judgments than direct quality rating by GPT-4. While these results are promising, the tasks studied (like topic-based summarization and boolean satisfaction problems) were relatively simple compared to more complex problems where flaws are more open-ended, and require so-

*These authors contributed equally.

phisticated reasoning or deep domain expertise to identify. In order to scale critique-based oversight to such domains, we need datasets containing not just examples of correct solutions, but also examples of flawed solutions where both the presence and specific nature of the errors are known and validated by domain experts. This would extend the approach of using synthetic tasks with hand-coded oracles to more sophisticated settings, allowing researchers to directly measure whether models' critiques align with actual expert-identified flaws in complex reasoning.

Debate is an approach that aims to enhance a weak judge's ability to evaluate claims and determine truth by leveraging the power of two strong AI agents engaging in a structured debate about the correct solution to a given problem (Irving, Christiano, and Amodei 2018). Early empirical results from debate experiments have demonstrated that agents trained in debate settings can achieve higher accuracy in identifying truthful claims compared to settings where they view output from a single agent randomly assigned to advocate for a correct or incorrect answer (Kenton et al. 2024; Khan et al. 2024; Michael et al. 2023), and that judgment accuracy scales with the capabilities of the debate models (keeping judge skill constant) (Kenton et al. 2024; Khan et al. 2024). However, this trend may break down as the difficulty of the problems that debaters are asked to consider increases, especially if the agents start relying more on heuristic reasoning and struggle to fully articulate their decision-making process.

One approach to measure debate's effectiveness as question difficulty increases is to conduct multi-domain debate studies in the vein of Kenton et al. (2024) with datasets of varying difficulty levels to track performance degradation. However, a decline in performance could stem from two distinct mechanisms: (1) judges might struggle more with evaluating debates, but without making systematic errors that incentivize debater dishonesty, or (2) agents on both sides of the debate might increasingly attempt to convince judges of incorrect or questionable claims; we would expect this to occur if arguing on the side of the truth provides decreasing advantage as question difficulty increases. Datasets containing information about the presence and nature of flaws in solutions to challenging questions would enable researchers to distinguish between the two scenarios by directly testing whether debate models are identifying genuine flaws or persuading judges with specious arguments.

Prover-verifier games are a game-theoretic framework involving two agents, a powerful but untrusted prover and a computationally limited but trusted verifier (Anil et al. 2021). The prover's role is to generate a "proof" (or justification) for a decision (for example, a classification) while the verifier must independently check the provided evidence. Kirchner et al. (2024) extended this setup by introducing a training regime that explicitly encourages models to support their generated solutions with reasoning that can be accurately checked by human judges or smaller verifiers, a property they call legibility. Brown-Cohen, Irving, and Piliouras (2025) apply a similar framework to debate to address shortcomings identified in prior work.

Incorporating detailed error annotations could be relevant to studies of the real-world usefulness of prover-verifier games in two ways. First, carefully curated datasets containing correct and flawed solutions for difficult problems on diverse topics could provide empirical evidence on how well a given verifier generalizes to challenging tasks across a wide range of domains. Second, if future work shows that human legibility decreases as problem difficulty increases, then distinguishing whether or not this decline is due to the helpful prover increasingly resorting to specious arguments will be crucial for improving the scaling properties of prover-verifier games as an oversight protocol.

Our contribution. Well-curated datasets containing both detailed correct solutions validated by domain experts and flawed solutions with annotated errors have the potential to serve an important role in evaluations of scalable oversight methods, allowing the research community to track progress over time and ensure that improvements seen in controlled experiments extend to more challenging problems. FindTheFlaws takes first steps towards addressing this need by providing diverse, expert-validated datasets that enable analysis of how well different oversight approaches enable humans or weak models to determine whether solutions to challenging problems are flawed. By including information about the nature of errors in flawed solutions in addition to binary correctness labels, our datasets enable researchers to assess whether oversight methods are reliably identifying the actual flaws in flawed solutions as opposed to developing heuristics that will not scale well with problem difficulty.

2 Related Work

Scalable oversight. Ensuring reliable oversight of AI systems becomes increasingly challenging as these systems tackle tasks whose solutions are difficult to verify. Bowman et al. (2022) investigated the performance of a simple baseline—direct interaction with an LLM—using MMLU (Hendrycks et al. 2021) and QuALITY (Pang et al. 2022). Despite the approach's simplicity, they found that human judges interacting with models achieved higher performance than either human judges or models alone. Other approaches to scaling AI supervision include self-critique (Saunders et al. 2022), debate (Irving, Christiano, and Amodei 2018; Brown-Cohen, Irving, and Piliouras 2025), prover-verifier games (Anil et al. 2021; Kirchner et al. 2024), market-making (Hubinger 2020), and recursive reward modeling (Leike et al. 2018). More background information on scalable oversight can be found in Grey et al. (2025).

LLM critique/evaluation benchmarks. While FindTheFlaws focuses on expert-annotated solutions with and without flaws to test models' verification abilities, benchmarks such as CriticEval (Lan et al. 2025) focus on testing LLMs' abilities to provide textual critiques and scalar-valued judgements of LLM outputs. Benchmarks with similar goals include CriticBench (Lin et al. 2024), MetaCritique (Sun et al. 2024a), SummEval (Fabbri et al. 2021), and WMT-22 (Freitag et al. 2022), although these cover a narrower range of tasks or domains, as well as MT-Bench (Zheng et al. 2023), which addresses the evaluation of multi-turn conversational ability and instruction-following. There has also been work

on error identification in process-based settings for mathematical reasoning (Zheng et al. 2025; Song et al. 2025; Zeng et al. 2024a), meta-reasoning in scientific and logical tasks (Zeng et al. 2024b), and PaLM-2 errors in logical and arithmetic tasks (Tyen et al. 2024). In contrast, FindTheFlaws covers a wider range of domains with difficult questions, contains frontier model and expert-generated flaws, and has expert annotations for the location and nature of errors in model reasoning.

Hallucination benchmarks. Various benchmarks have been crafted to test LLMs on hallucinations, factual mistakes, and reasoning flaws hidden in otherwise plausible responses (Lin, Hilton, and Evans 2022; Lee et al. 2022; Min et al. 2023; Yin et al. 2023; Li et al. 2023; Muhlgay et al. 2024). By the broad criteria outlined in a recent survey by Huang et al. (2025), FindTheFlaws could be considered a hallucination detection benchmark that annotates both factuality and faithfulness hallucinations. However, its focus on errors that occur in answers to challenging questions and require expert analysis to detect distinguish it from other hallucination detection benchmarks enumerated in Huang et al. (2025). SelfCheckGPT-Wikibio (Muennighoff et al. 2024), HaluEval (Li et al. 2023), and most other hallucination detection benchmarks in their survey consider faithfulness hallucinations only. The FELM factuality benchmark (Zhao et al. 2023) covers multiple domains, annotates errors of factuality as well as faithfulness, and provides error locations and explanations. The combination of all of these characteristics makes it most comparable to CELS, one of the five datasets that comprise FindTheFlaws. However, FindTheFlaws is a substantially larger dataset and focuses specifically on difficult questions requiring domain-specific expertise to answer, as well as on outputs which include errors that are egregious enough as to result in an incorrect final conclusion, two characteristics highly relevant to scalable oversight research.

Synthetic flaw generation. In the prover-verifier setup of Kirchner et al. (2024), a "sneaky" prover is trained to generate incorrect yet convincing solutions. Other researchers have explored methods for training language models to generate text with undesirable properties that evade detectors (Perez et al. 2022). The flawed examples in FindTheFlaws were either identified through adversarial selection, or generated by prompting models to introduce flaws which were manually reviewed and sometimes improved upon or rewritten by human experts. In addition, our expert-curated flaws in GPQA Diamond Plus may explore different areas of the space of possible errors than adversarial training alone.

3 Methods

In this section, we present the tasks we use to evaluate error-detection capabilities of models using our datasets. FindTheFlaws contains novel modifications to existing datasets (Chen et al. 2023; Rein et al. 2023; Puri et al. 2021; Jin et al. 2021) and a completely new dataset with sentence-level expert annotations on model-generated long-form reasoning in various domains. Each dataset contains questions with their correct final answers, one or more long-form 'solutions' labeled as either correct or flawed, and information about the nature (location and/or description) of errors in flawed solutions. We provide a summary of our datasets and the number of samples in each in Table 1, and we include dataset construction details in the extended version of this paper linked on the title page.

We evaluate error detection capabilities of *gpt-4o-2024-11-20*, *o3-mini-2025-01-31 (medium)*, *o3-2025-04-16 (high)*, *o4-mini-2025-04-16 (high)*, *claude-3-5-sonnet-20241022*, *claude-3-7-sonnet-20250219*, *claude-opus-4-20250514*, and Llama 3.3 70B models using all 'reliable' samples (details in the extended version) from our datasets with the Inspect evaluation framework (0.3.114) on Windows 11. The framework and documentation is offered by the UK AI Security Institute (2024). For all datasets, we create two tasks for the models to be evaluated on, of (1) whether the model's assessment of whether a long-form reasoning solution is correct or flawed matches the ground truth, and (2) if the solution is flawed, whether it can identify the specific error in the flawed solution. Although the general capabilities tested by our evaluations remain the same, the task setup varies across the datasets. The following sections describe how these tasks have been implemented for all categories of datasets.

Solution-level tasks Our basic evaluation setup consists of a question and a proposed solution where we have ground truth information about whether the solution is 'CORRECT' or 'FLAWED', and expert annotations identifying errors in all 'FLAWED' solutions. The model being evaluated is prompted to judge the reasoning of the proposed solution and classify it as 'CORRECT' or 'FLAWED', and to identify any errors it finds in the solution (see extended version for prompt templates). The model output is then used to evaluate performance on the following tasks:

- **Match**: Does the model's assessment of whether a long-form reasoning solution is 'CORRECT' or 'FLAWED' match the ground truth?
- **Error-grading**: If the solution is marked 'FLAWED' by the ground truth, is the error identified by the model equivalent to the error identified in a human expert's judgment of the solution?

For the error-grading task, we use Claude 3.5 Sonnet (*claude-3-7-sonnet-20250219*) to classify whether the judgment generated by the model is equivalent to the ground truth judgment made by a human expert using an error-grading prompt. For the Modified TheoremQA and GPQA Diamond Plus datasets, we evaluate whether the model can identify the first error in the solution. For the Adversarial MedQA dataset, we evaluate whether expert descriptions of the nature of the error identify the same problems as the model. When we have error descriptions from multiple annotators for Adversarial MedQA samples, we only use the primary annotator's description as the ground truth. For reproducibility, all random seeds are set to constants; code, datasets, and full evaluation output is available at the "Code and Datasets" link on the title page.

Python650 tasks We have broken down the Python650 dataset into three subsets based on the type of long-form rea-

Dataset Name	Description	$N_{reliable}$	N_{total}
Adversarial MedQA	LLM-generated and expert annotated solutions to adversarially selected questions from MedQA (Jin et al. 2021).	222	319
GPQA Diamond Plus	Stepwise correct solutions and expert-written flawed solutions to questions from GPQA Diamond (Rein et al. 2023).	382	396
Modified TheoremQA	LLM-generated stepwise correct and flawed solutions to questions from TheoremQA (Chen et al. 2023).	190	190
Python650	LLM-generated and expert-annotated arguments for and against proposed solutions to Python800 (Puri et al. 2021) questions.	1266	1296
CELS Law	Sets of four LLM-generated solutions (13 sentences per solution) to US contract law and evidence law questions along with sentence-level expert annotations and critiques.	40 (522)	40 (522)
CELS Lojban	Sets of four LLM-generated solutions (10 sentences per solution) to Lojban questions along with sentence-level expert annotations and critiques.	120 (1218)	192 (1938)
CELS Surgery	Sets of four LLM-generated solutions (11 sentences per solution) to surgery questions along with sentence-level expert annotations and critiques.	212 (2282)	220 (2383)

Table 1: List of datasets in included in FindTheFlaws, along with number of samples marked as reliable (questions and choices verified to be unambiguous and ready to use) by expert annotators. We report the total number of individually annotated sentences for the CELS datasets in parentheses.

soning that the model is judging:

- **Standard Python650**: This setup contains a question and a proposed solution, similar to the setup in Section 3. Due to the lack of ground truth data identifying the nature of the errors, we only evaluate the match task for this subset.
- **Meta Python650**: This setup contains a question, a proposed solution, and an argument regarding the correctness of the proposed solution. The model being evaluated needs to check if the argument accurately describes the correctness of the proposed solution (match task), and to produce an explanation of why or why not. For cases in which the argument does not accurately describe the correctness of the solution, the model's explanation is compared to ground truth explanations made by human experts about the problems with the argument (error-grading task).
- **Alt Meta Python650**: This setup is similar to Meta Python650, but we first filter the samples so that we only evaluate flawed arguments that accurately classify the proposed solution as correct or incorrect, but which do not correctly identify the actual errors in the solution. Performance on the match task is not reported, as all samples in this set were pre-selected to correctly classify solutions as 'CORRECT' or 'FLAWED'.

We use the match and error-grading tasks described in Section 3 to evaluate model performance on the three subsets described above. Whenever we have multiple annotator explanations for a sample, we only use a single explanation as the ground truth for the error-grading task.

CELS tasks The CELS dataset contains expert annotations for each sentence in the proposed solution, so we adapt the tasks mentioned in Section 3 to sentence level labels and judgments as follows:

- **Match-all**: Does the model correctly classify sentences as 'CORRECT' or 'FLAWED' compared to ground truth labels?
- **Grade-all**: Does the model identify errors in 'FLAWED' sentences that are equivalent to errors described in expert judgments for each sentence?

Sentences that are not classified as 'CORRECT' or 'FLAWED' by a majority of expert annotators are excluded from evaluation. We treat each sentence as a sample for the evaluations, but we take into account the clustering introduced by reusing the same question for different sentences when we report scores in Section 4.3. Each evaluation was conducted via a single run, i.e., there was no use of self-consistency (Wang et al. 2023).

4 Results

We present the results of our evaluations for all datasets grouped according to the tasks described in Section 3. Our figures in this section only present *F1 score* for match tasks (treating correct solutions as the positive class) and *accuracy* for error-grading tasks (the percentage of flawed solutions where the model accurately identified the error in the solution). Table 2 contains the best model performance compared to baselines for all tasks we tested. Please refer to the extended version of this paper for all metrics calculated to capture model performance and for detailed descriptions of the expert baselines used for each task.

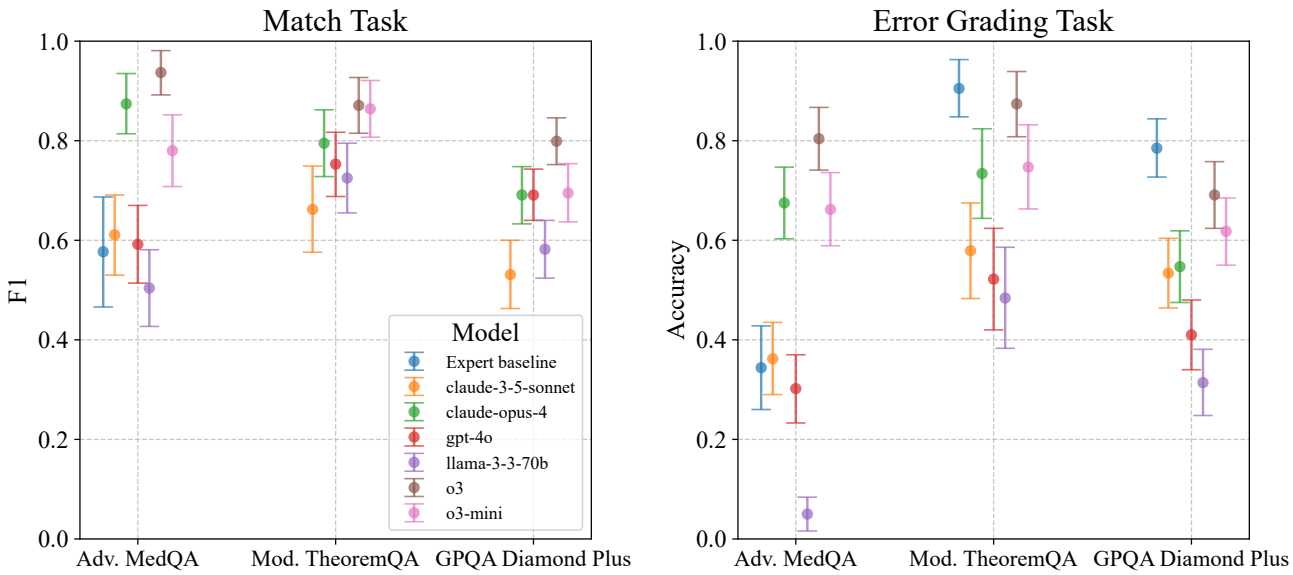

Figure 1: Performance of selected models, and expert baselines, on match/grading metrics for Adversarial MedQA, Modified TheoremQA, and GPQA Diamond Plus. Claude 3.7 Sonnet and o4-mini (not displayed) are consistently slightly worse at the tasks than Claude Opus 4 and o3 respectively. Expert baselines for Adversarial MedQA represent the performance of a human clinician, while baselines for the other two datasets (error-grading only) represent agreement between o3 and the solution authors about the location of the first error when o3 is provided with the labeled correct and flawed solutions developed by the solution authors. 95% confidence intervals were calculated using a cluster-based block bootstrap approach (1,000 samples).

4.1 Solution-level results

We present evaluation results for the match and error-grading tasks described in Section 3 for the Adversarial MedQA, Modified TheoremQA and GPQA Diamond Plus datasets in Figure 1. We find that the performance on the match task is relatively close for the top models, but that there is more variability on error-grading. In particular, GPT-4o's performance relative to other models is much poorer on the latter task. This highlights that there can be a disconnect between a model's proficiency in identifying correct solutions and its capacity to accurately characterize errors.

We also provide a baseline for Adversarial MedQA by reporting the performance of a single human expert at the same tasks given to the models being evaluated. We find that almost all models outperform individual human experts in both tasks, but it is possible that a stronger baseline derived from the views of multiple clinicians (as was used to create the ground truth for Adversarial MedQA) could have achieved higher performance.

We do not have human baselines for the other two datasets in this section. Instead, we create expert baseline analogs for the error-grading task by providing o3 with the correct and flawed solutions and prompting it to describe the first error in the flawed solution. This description is compared to the ground truth explanations used in our evaluations in the error-grading task to check if both mention the same first error. The differences between model performance and these baseline analogs suggest that models miss certain errors when judging flawed solutions which they are able to find when provided with correct information. This suggests that capabilities for error detection in long-form science and math solutions still have room for improvement. More details about the rationale, construction, and interpretation of these expert baselines and baseline analogs can be found in the extended version.

4.2 Python650 results

We present the results for the Python650 evaluations in Figure 2. We do not report error-grading results for the standard Python650 dataset, due to a lack of ground truth arguments identifying issues with the solutions. However, match and error-grading tasks can both be computed for Meta Python650.

We find that o3 achieves extremely high scores in the match task for both datasets, even exceeding the expert baseline for Meta Python650. We observe lower performance from all models on the error-grading task. Given the low expert baseline on this task, however, this discrepancy may simply reflect the inherent difficulty of achieving consistent error descriptions across evaluators, rather than an intrinsic limitation in model capabilities.

Claude Opus 4 and o3 exceed baseline performance on the error-grading task for the Meta Python650 dataset, followed by Claude 3.7 Sonnet and o3-mini matching baseline performance. The model and baseline performance drops significantly for the Alt Meta Python650 dataset, but the relative performance of the models follows the same trends. Manual inspection of common failure modes on this dataset sug-

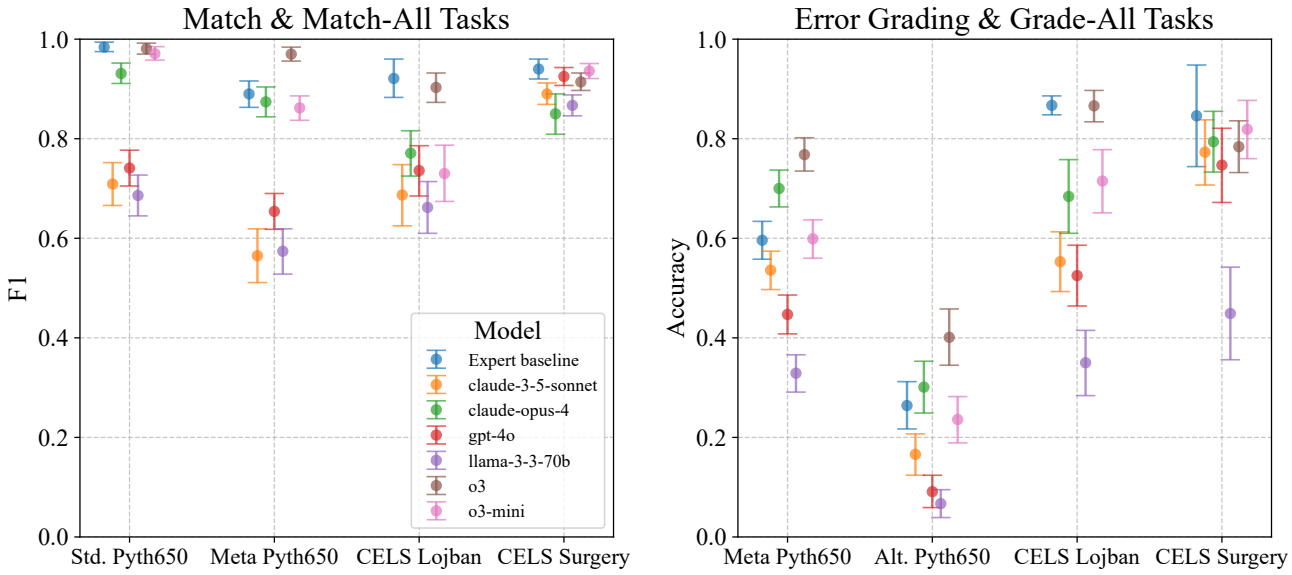

Figure 2: Performance of selected models, as well as human expert baselines, on match task for Python650 and Meta-Python650, on the error-grading task for two subsets of Meta-Python650, and sentence-level CELS Surgery and Lojban tasks. Expert baselines represent the performance of a single human expert for CELS Lojban, and of a majority vote of three clinicians for CELS Surgery. Claude 3.7 Sonnet and o4-mini (not displayed) are consistently slightly worse at the tasks than Claude Opus 4 and o3 respectively. 95% confidence intervals were calculated using a cluster-based block bootstrap approach (1,000 samples).

gested that models and annotators alike frequently missed subtle errors in arguments that had issues but ultimately argued for the correct classification, suggesting that it is easier for models to accurately find flaws in arguments that have an incorrect final answer, compared to arguments with flawed reasoning but a correct final answer.

4.3 CELS results

We present the evaluation results for the match-all (F1) and grade-all (Accuracy) tasks for CELS Lojban and CELS Surgery in Figure 2. The tasks track sentence-level error detection capabilities of models, and the baselines have been created using expert annotators performing the tasks for 20 arguments in each dataset (annotating each sentence in each of these). Further details about the expert baseline are provided in the extended version.

Results for CELS Law are not shown due to space constraints and concerns about comparability with the other datasets in the figure (due to its much smaller size), but these are reported in the extended version of the paper. In short, the Claude family and o3 outperform all other models on CELS Law, but error bars are very large due to the small dataset size. Additionally, o3-mini performs unusually poorly at CELS Law.

We find that model performance on the match-all task is more clustered than for other datasets, where o3 and the Claude family are frequently standout performers. Additionally, we see that the expert baseline is much higher than model performance for CELS Lojban (only matched by o3), unlike for CELS Surgery. The grade-all task performance varies across datasets, with the CELS Lojban expert baseline well above all models except o3, performance being similar across the board for CELS Surgery.

5 Discussion

5.1 Key findings

Our evaluation of frontier language models on FindTheFlaws yielded several observations. Model performance on the match task (distinguishing correct from flawed solutions) did not always reflect performance on error-grading tasks. For example, while Claude Opus 4, o3-mini, and GPT-4o all performed comparably well on the match task for GPQA Diamond Plus, GPT-4o's performance was lower on the error-grading task while Claude Opus 4 and o3-mini maintained strong performance. This suggests that the ability to recognize whether a solution contains an error is distinct from the more demanding capability of identifying and explaining the specific nature of that error. This distinction is particularly relevant for scalable oversight protocols in which the goal is sometimes to enable an overseer to identify particular flaws rather than merely to detect their presence.

We observed that there were some task/dataset combinations on which performance differed substantially across frontier models. These have special utility for approaches to scalable oversight evaluation such as 'sandwiching' with LLM-based evaluators (Pung and Mukobi 2023; Kenton et al. 2024; Khan et al. 2024; Arnesen, Rein, and Michael

Subset	Match (F1-score)		Error-grading (Accuracy)		$N_{samples}$ (CORRECT, FLAWED)	
	Model	Baseline	Model	Baseline	Model	Baseline
Adversarial MedQA	0.937 0.892–0.981	0.548 0.469–0.627	0.804 0.741–0.867	0.344 0.260–0.428	(62, 160)	(62, 160)
GPQA Diamond Plus	0.799 0.752–0.846	–	0.691 0.624–0.758	0.785 0.727–0.844	(191, 191)	(0, 191)
Modified TheoremQA	0.872 0.819–0.925	–	0.874 0.808–0.939	0.905 0.848–0.963	(95, 95)	(0, 95)
Standard Python650	0.981 0.970–0.992	0.984 0.975–0.994	–	–	(316, 317)	(316, 317)
Meta Python650	0.970 0.956–0.984	0.890 0.863–0.916	0.768 0.735–0.802	0.596 0.558–0.634	(316, 613)	(319, 633)
Alt. Meta Python650	–	–	0.401 0.741–0.867	0.264 0.217–0.312	(0, 294)	(0, 314)
CELS Law	0.901 0.875–0.927	0.907 0.862–0.952	0.819 0.746–0.892	0.587 0.481–0.693	(200, 196)	(223, 196)
CELS Lojban	0.903 0.873–0.932	0.921 0.883–0.960	0.866 0.834–0.897	0.867 0.848–0.886	(399, 648)	(83, 98)
CELS Surgery	0.936 0.921–0.951	0.940 0.920–0.960	0.819 0.760–0.877	0.846 0.744–0.948	(1245, 568)	(129, 39)

Table 2: Best model performance metrics for all tasks tested in this paper. The match task was run using both 'CORRECT' and 'FLAWED' samples and the error-grading task was run using only 'FLAWED' samples. The CELS results are for the match-all and grade-all tasks. 95% confidence intervals were calculated using a cluster-based block bootstrap approach (1,000 samples). 'CORRECT' = positive samples, 'FLAWED' = negative samples.

2024; Sun et al. 2024b), which requires gold labels, a weak LLM to play the role of evaluator, and an LLM whose capabilities exceed those of the evaluator but fall short of perfect performance, to act as the system being overseen. For example, the match task for Python650 and Meta Python650, and the error-grading task for CELS Lojban, Meta Python650, and Adversarial MedQA, all are examples of tasks for which the expert baseline or the top model outperforms at least one model, which is in turn underperformed by yet another model. Claude Opus 4 and o4-mini generally demonstrated strong performance, but o3 clearly led the pack distinguishing correct from flawed solutions and grading errors for Adversarial MedQA, GPQA, Python650, and Meta-Python650. In contrast, Claude models showed strong performance on the error-grading task in CELS Law, where o3-mini and GPT-4o performed very poorly.

We find that for several datasets, particularly Adversarial MedQA, top-performing models consistently match or outperform human experts. Notably, CELS Lojban stands as an exception, with all models except o3 performing significantly below human expert baselines. Finally, our results highlight the difficulty of the Alt Meta Python650 error-grading task, where models must identify flaws in explanations that reach the correct conclusion through faulty reasoning. This represents a particularly challenging form of verification, although the low degree of agreement among human experts also suggests this task may involve inherent ambiguities that make consensus difficult even for skilled programmers.

5.2 Implications for scalable oversight

FindTheFlaws was developed to address a gap in existing AI evaluation resources: the lack of datasets containing not just examples of correct solutions to challenging problems, but also flawed solutions where both the presence and specific nature of errors are known and validated by domain experts.

Even in cases where current models exceed human performance on certain tasks, these datasets remain valuable for at least three reasons. First, they provide a diverse testbed for evaluating whether oversight mechanisms can help weak verifiers effectively evaluate outputs from more capable systems. Second, they enable researchers to analyze whether oversight methods are identifying genuine errors or developing heuristics that may not generalize well with increasing problem difficulty. Third, they allow performance baselines for scalable oversight methods to be established across diverse domains that can help track progress as new models and oversight techniques are developed.

The finding that models sometimes outperform human experts, particularly in the Adversarial MedQA dataset, highlights the fundamental challenge motivating scalable oversight research: as AI systems tackle increasingly specialized and complex tasks, naive approaches to human verification become not only more costly but also less able to continue providing training signals that we can have confidence in. The growing number of domains in which AI systems match, exceed, or are quickly approaching human expert performance suggests we may already be entering the regime where the necessity for scalable oversight approaches is not merely theoretical but practical and imme-

diate.

Finally, the gap between match and error-grading performance for some models highlights that the ability (or tendency) to determine whether a solution is correct is not always accompanied by the successful identification of the specific nature of the errors. If assistant models within a particular scalable oversight protocol are more likely to present flawed reasoning to judges when discussing questions that are more difficult for the judge, this suggests potential scaling limitations that might otherwise remain hidden until more challenging domains are encountered.

5.3 Limitations and future work

Several limitations of our current approach warrant consideration. First, the process of introducing flaws into solutions inevitably involves some degree of artificiality. While some errors exist due to limits in model capabilities (e.g., all errors in Adversarial MedQA) and we took care to ensure errors were validated by domain experts, the distribution of error types may differ from those occurring in the output of deceptive models, models trained in debate, etc. Future dataset expansions could aim to make the distribution of errors more representative of what might be expected from a scalable oversight protocol involving adversarial training. Future exploration of the long CoT regime (Chen et al. 2025) will also be important for elicitation with stronger models.

Our evaluation methodology, which uses Claude 3.5 Sonnet to determine whether model-identified errors match expert annotations, introduces another potential source of bias. While this approach enabled efficient evaluation across large datasets, it may systematically favor certain error descriptions or explanation styles. Model performance is also likely to be influenced by the specific prompt formulations used in the evaluation. Subtle changes in how verification tasks are framed could impact model responses, raising concerns about robustness. Future research should explore prompt engineering techniques that minimize sensitivity to phrasing and improve the consistency of model performance.

The challenge of obtaining reliable expert judgments is evident in the varying levels of inter-annotator agreement across datasets. The expert baseline analogues created for GPQA Diamond Plus and Modified TheoremQA highlight the inherent ambiguity in identifying the locations of errors even when both the correct answers and flawed answers are known. This underscores the importance of multi-annotator approaches and careful validation.

Finally, as previously noted, current frontier models already perform at or above our conservative estimates of human expert performance on most of the datasets in FindTheFlaws. While these datasets remain useful for testing scalable oversight protocols involving weaker judges, benchmarks with error annotations in even more challenging domains would facilitate improved evaluation of whether oversight mechanisms that appear effective today will continue to be reliable as AI capabilities advance.

6 Conclusion

The FindTheFlaws datasets offer a resource enabling researchers to evaluate oversight techniques by testing how effectively model-assisted evaluators can spot errors in long-form reasoning across various complex domains. For some models, our analysis revealed a discrepancy between the model's ability to detect the presence of errors and its capacity to identify or explain those errors. This distinction is relevant to oversight protocols in which the ability of the protocol to scale effectively with problem difficulty is linked to its ability to enable overseers to accurately identify specific flaws that may appear in invalid or deceptive reasoning. On several task/dataset combinations, Llama 3.3 70B was the weakest model, and there was at least one additional model with performance that clearly exceeded Llama's but fell short of the human expert baseline or best model. This implies that these task/dataset combinations are well-suited for experiments which require gold labels or a strong expert model, a weaker model acting as judge, and a model of intermediate performance to serve as the system being overseen.

Our findings also highlight the domain-specific nature of model performance in oversight tasks, with relative error-grading performance for models like o3-mini and Claude 4 Opus flipping between coding and scientific reasoning. These domain-specific differences underscore the importance of evaluating oversight techniques across diverse problem domains rather than relying on performance in a single area. As AI capabilities continue to advance, robust oversight protocols will become increasingly critical, and we hope that FindTheFlaws will serve as a valuable resource for this continuing effort.

Acknowledgments

We thank Nora Petrova and Monika Jotautaitė for helpful discussions. Support for this project was provided by Open Philanthropy and the Good Ventures Foundation.

References

Amodei, D.; Olah, C.; Steinhardt, J.; Christiano, P.; Schulman, J.; and Mané, D. 2016. Concrete Problems in AI Safety. arXiv:1606.06565.

Anil, C.; Zhang, G.; Wu, Y.; and Grosse, R. 2021. Learning to Give Checkable Answers with Prover-Verifier Games. arXiv:2108.12099.

Arnesen, S.; Rein, D.; and Michael, J. 2024. Training Language Models to Win Debates with Self-Play Improves Judge Accuracy. arXiv:2409.16636.

Bowman, S. R.; Hyun, J.; Perez, E.; Chen, E.; Pettit, C.; Heiner, S.; Lukošiūtė, K.; Askell, A.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; Olah, C.; Amodei, D.; Amodei, D.; Drain, D.; Li, D.; Tran-Johnson, E.; Kernion, J.; Kerr, J.; Mueller, J.; Ladish, J.; Landau, J.; Ndousse, K.; Lovitt, L.; Elhage, N.; Schiefer, N.; Joseph, N.; Mercado, N.; DasSarma, N.; Larson, R.; McCandlish, S.; Kundu, S.; Johnston, S.; Kravec, S.; Showk, S. E.; Fort, S.; Telleen-Lawton, T.; Brown, T.; Henighan, T.; Hume, T.; Bai, Y.; Hatfield-Dodds, Z.; Mann, B.; and Kaplan, J. 2022. Measuring Progress on Scalable Oversight for Large Language Models. arXiv:2211.03540.

Brown-Cohen, J.; Irving, G.; and Piliouras, G. 2025. Avoiding Obfuscation with Prover-Estimator Debate. arXiv:2506.13609.

Chen, Q.; Qin, L.; Liu, J.; Peng, D.; Guan, J.; Wang, P.; Hu, M.; Zhou, Y.; Gao, T.; and Che, W. 2025. Towards Reasoning Era: A Survey of Long Chain-of-Thought for Reasoning Large Language Models. arXiv:2503.09567.

Chen, W.; Yin, M.; Ku, M.; Lu, P.; Wan, Y.; Ma, X.; Xu, J.; Wang, X.; and Xia, T. 2023. TheoremQA: A Theorem-Driven Question Answering Dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 7889–7901.

Christiano, P.; Shlegeris, B.; and Amodei, D. 2018. Supervising strong learners by amplifying weak experts. arXiv:1810.08575.

Fabbri, A. R.; Kryściński, W.; McCann, B.; Xiong, C.; Socher, R.; and Radev, D. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9: 391–409.

Freitag, M.; Rei, R.; Mathur, N.; Lo, C.-k.; Stewart, C.; Avramidis, E.; Kocmi, T.; Foster, G.; Lavie, A.; and Martins, A. F. 2022. Results of WMT22 metrics shared task: Stop using BLEU–neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 46–68.

Grey, M.; et al. 2025. AI Safety Atlas. French Center for AI Safety (CeSIA).

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. arXiv:2009.03300.

Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2): 1–55.

Hubinger, E. 2020. AI Safety via Market Making.

Irving, G.; Christiano, P.; and Amodei, D. 2018. AI safety via debate. arXiv:1805.00899.

Jin, D.; Pan, E.; Oufattole, N.; Weng, W.-H.; Fang, H.; and Szolovits, P. 2021. What disease does this patient have? A large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14): 6421.

Kenton, Z.; Siegel, N.; Kramár, J.; Brown-Cohen, J.; Albanie, S.; Bulian, J.; Agarwal, R.; Lindner, D.; Tang, Y.; Goodman, N.; et al. 2024. On scalable oversight with weak LLMs judging strong LLMs. *Advances in Neural Information Processing Systems*, 37: 75229–75276.

Khan, A.; Hughes, J.; Valentine, D.; Ruis, L.; Sachan, K.; Radhakrishnan, A.; Grefenstette, E.; Bowman, S. R.; Rocktäschel, T.; and Perez, E. 2024. Debating with more persuasive LLMs leads to more truthful answers. In *ICML 2024 (Best Paper Award)*.

Kirchner, J. H.; Chen, Y.; Edwards, H.; Leike, J.; McAleese, N.; and Burda, Y. 2024. Prover-Verifier Games improve legibility of LLM outputs. arXiv:2407.13692.

Lan, T.; Zhang, W.; Xu, C.; Huang, H.; Lin, D.; Chen, K.; and Mao, X.-L. 2025. CriticEval: Evaluating large-scale language model as critic. *Advances in Neural Information Processing Systems*, 37: 66907–66960.

Lee, N.; Ping, W.; Xu, P.; Patwary, M.; Fung, P. N.; Shoeybi, M.; and Catanzaro, B. 2022. Factuality Enhanced Language Models for Open-Ended Text Generation. *Advances in Neural Information Processing Systems*, 35: 34586–34599.

Leike, J.; Krueger, D.; Everitt, T.; Martic, M.; Maini, V.; and Legg, S. 2018. Scalable agent alignment via reward modeling: a research direction. arXiv:1811.07871.

Li, J.; Cheng, X.; Zhao, W. X.; Nie, J.-Y.; and Wen, J.-R. 2023. HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models. arXiv:2305.11747.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Lin, S.; Hilton, J.; and Evans, O. 2022. TruthfulQA: Measuring How Models Mimic Human Falsehoods. arXiv:2109.07958.

Lin, Z.; Gou, Z.; Liang, T.; Luo, R.; Liu, H.; and Yang, Y. 2024. CriticBench: Benchmarking LLMs for Critique-Correct Reasoning. arXiv:2402.14809.

Michael, J.; Mahdi, S.; Rein, D.; Petty, J.; Dirani, J.; Padmakumar, V.; and Bowman, S. R. 2023. Debate Helps Supervise Unreliable Experts. arXiv:2311.08702.

Min, S.; Krishna, K.; Lyu, X.; Lewis, M.; tau Yih, W.; Koh, P. W.; Iyyer, M.; Zettlemoyer, L.; and Hajishirzi, H. 2023. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. arXiv:2305.14251.

Muennighoff, N.; Hongjin, S.; Wang, L.; Yang, N.; Wei, F.; Yu, T.; Singh, A.; and Kiela, D. 2024. Generative representational instruction tuning. In *ICLR 2024 Workshop: How Far Are We From AGI*.

Muhlgay, D.; Ram, O.; Magar, I.; Levine, Y.; Ratner, N.; Belinkov, Y.; Abend, O.; Leyton-Brown, K.; Shashua, A.; and Shoham, Y. 2024. Generating Benchmarks for Factuality Evaluation of Language Models. arXiv:2307.06908.

Pang, R. Y.; Parrish, A.; Joshi, N.; Nangia, N.; Phang, J.; Chen, A.; Padmakumar, V.; Ma, J.; Thompson, J.; He, H.; and Bowman, S. R. 2022. QuALITY: Question Answering with Long Input Texts, Yes! arXiv:2112.08608.

Perez, E.; Huang, S.; Song, F.; Cai, T.; Ring, R.; Aslanides, J.; Glaese, A.; McAleese, N.; and Irving, G. 2022. Red Teaming Language Models with Language Models. arXiv:2202.03286.

Pung, S.; and Mukobi, G. 2023. Automated Sandwiching: Efficient Self-Evaluations of Conversation-Based Scalable Oversight Techniques. Accepted at the ScaleOversight research sprint on February 16, 2023. Accessed: 2025-03-12.

Puri, R.; Kung, D. S.; Janssen, G.; Zhang, W.; Domeniconi, G.; Zolotov, V.; Dolby, J.; Chen, J.; Choudhury, M. R.; Decker, L.; Thost, V.; Buratti, L.; Pujar, S.; and Finkler, U. 2021. Project CodeNet: A Large-Scale AI for Code Dataset for Learning a Diversity of Coding Tasks. *CoRR*, abs/2105.12655.

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; and Bowman, S. R. 2023. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. arXiv:2311.12022.

Saunders, W.; Yeh, C.; Wu, J.; Bills, S.; Ouyang, L.; Ward, J.; and Leike, J. 2022. Self-critiquing models for assisting human evaluators. arXiv:2206.05802.

Song, M.; Su, Z.; Qu, X.; Zhou, J.; and Cheng, Y. 2025. PRMBench: A Fine-grained and Challenging Benchmark for Process-Level Reward Models. arXiv:2501.03124.

Sun, S.; Li, J.; Yuan, W.; Yuan, R.; Li, W.; and Liu, P. 2024a. The Critique of Critique. arXiv:2401.04518.

Sun, Z.; Yu, L.; Shen, Y.; Liu, W.; Yang, Y.; Welleck, S.; and Gan, C. 2024b. Easy-to-Hard Generalization: Scalable Alignment Beyond Human Supervision. In *Advances in Neural Information Processing Systems (NeurIPS 2024)*.

Tyen, G.; Mansoor, H.; Cărbune, V.; Chen, P.; and Mak, T. 2024. LLMs cannot find reasoning errors, but can correct them given the error location. arXiv:2311.08516.

Uesato, J.; Kushman, N.; Kumar, R.; Song, F.; Siegel, N.; Wang, L.; Creswell, A.; Irving, G.; and Higgins, I. 2022. Solving math word problems with process- and outcome-based feedback. arXiv:2211.14275.

UK AI Security Institute. 2024. Inspect AI: Framework for Large Language Model Evaluations. https://inspect.ai-safety-institute.org.uk/. Accessed: May 4, 2025.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. arXiv:2203.11171.

Yin, Z.; Sun, Q.; Guo, Q.; Wu, J.; Qiu, X.; and Huang, X. 2023. Do Large Language Models Know What They Don't Know? arXiv:2305.18153.

Zeng, Z.; Chen, P.; Liu, S.; Jiang, H.; and Jia, J. 2024a. MR-GSM8K: A Meta-Reasoning Benchmark for Large Language Model Evaluation. arXiv:2312.17080.

Zeng, Z.; Liu, Y.; Wan, Y.; Li, J.; Chen, P.; Dai, J.; Yao, Y.; Xu, R.; Qi, Z.; Zhao, W.; Shen, L.; Lu, J.; Tan, H.; Chen, Y.; Zhang, H.; Shi, Z.; Wang, B.; Guo, Z.; and Jia, J. 2024b. MR-Ben: A Meta-Reasoning Benchmark for Evaluating System-2 Thinking in LLMs. arXiv:2406.13975.

Zhao, Y.; Zhang, J.; Chern, I.; Gao, S.; Liu, P.; He, J.; et al. 2023. FELM: Benchmarking factuality evaluation of large language models. *Advances in Neural Information Processing Systems*, 36: 44502–44523.

Zheng, C.; Zhang, Z.; Zhang, B.; Lin, R.; Lu, K.; Yu, B.; Liu, D.; Zhou, J.; and Lin, J. 2025. ProcessBench: Identifying Process Errors in Mathematical Reasoning. arXiv:2412.06559.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623.