Methodological Challenges in Agentic Evaluations of AI Systems

Kevin Wei [*1] **Stephen Guth** [*1] **Gabriel Wu** [*1] **Patricia Paskov** [2]

Abstract

With the increased generality and advanced reasoning capabilities of AI systems, an increasing number of AI evaluations are *agentic evaluations*: evaluations involving complex tasks requiring environmental interaction, as opposed to knowledge-based question-answer benchmarks. However, no work has explored the methodological challenges of agentic evaluations or the practices necessary to ensure their validity, reliability, replicability, and efficiency. In this (work-in-progress) paper, we (1) define and formalize the agentic evaluation paradigm; (2) survey and analyze methodological problems in agentic evaluations; and (3) discuss the implications of agentic evaluations for AI governance. Our hope is to improve the state of agentic evaluations of AI systems, systematize the methodological work in this domain, and contribute to the establishment of a science of AI evaluations.

1. Introduction

The latest generation of AI systems—popularly termed "AI agents"—have demonstrated substantial advancements in reasoning, long-horizon planning, and the ability to engage with and act in complex environments using tools (Kwa et al., 2025; Chan et al., 2023; Qin et al., 2023; Boiko et al., 2023; Bengio et al., 2025). In addition to performing tasks such as information retrieval (e.g., answering user questions), these systems are now often claimed to be able to "independently accomplish tasks on your behalf" (OpenAI, 2025). As a result, the evaluation of these AI systems has required new datasets and methodologies that move beyond traditional knowledge-based question-answer (QA) evaluations (e.g., Wei et al. 2025a; Starace et al. 2025; Wijk et al. 2024).

We call this new approach *agentic evaluation*, which we view as a paradigm for AI evaluation that is distinct from the evaluation of AI agents in general.[1] Specifically, we formalize the definition of agentic evaluation as the evaluation of *compound AI systems* on *environmental tasks*. A combination of complexity in compound AI systems as well as the interactivity, task horizon, and large state and action spaces of agentic evaluation tasks adds substantial difficulty to agentic evaluation relative to multiple-choice question-answer (MCQA) evaluations (Toner et al., 2024). As a result of these same characteristics, agentic evaluations bear similarities to but nevertheless present new challenges in compared to evaluations in other contexts such as in reinforcement learning, in game-based settings, and in the social sciences.

Agentic evaluations are of great interest to stakeholders throughout the AI value chain. For the machine learning (ML) research community, robust agentic evaluations can measure progress in AI development. For policymakers, agentic evaluations can help anticipate broader societal impacts of AI such as labor disruption (Toner et al., 2024), and they may be crucial components of regulatory documents such as safety cases (Goemans et al., 2024).

Despite the importance of agentic evaluations, there has been no comprehensive discussion of the methodological challenges of this new paradigm. This review paper aims to fill that gap in the literature by making three key contributions:

1. We formalize the agentic evaluation paradigm.
2. We survey and analyze methodological problems, promising research directions, and related research areas in agentic evaluations.
3. We discuss the importance of agentic evaluations to AI governance and to the broader ML community.

2. Background

Prior work on AI agents and on agentic evaluation has focused primarily on the results of current datasets and on

[*] Equal contribution [1] Harvard University, Cambridge, MA, U.S. [2] Independent. Correspondence to: Kevin Wei <kevin-wei@acm.org>.

Workshop on Technical AI Governance (TAIG) at ICML 2025, Vancouver, Canada. Copyright 2025 by the author(s).

[1] The definition of "AI agent" is highly contested in the literature, and we will generally avoid using this term throughout the paper. See discussion in Appendix A.1.

benchmarks available for evaluation (e.g., Chang et al., 2024; Luo et al., 2025; Yehudai et al., 2025) but rarely discusses the shortcomings of existing evaluation methods. These reviews are also loosely scoped and often discuss QA-based evaluations of AI systems.

This article examines *agentic evaluation*, which we define as the evaluation of *compound AI systems* on *environmental tasks*. We define "compound AI systems" as systems that consist of at least one component in addition to a foundation model instance, e.g., scaffolding, tools, data sources, other model instances, etc.[2] We define "environmental tasks" as tasks that require interaction with an external environment, are under-specified (and thus require reasoning or planning to complete), are multi-step, and are quantitatively scoreable by an external party (excluding the user).

Our view is that agentic evaluation is a paradigm for evaluation and is limited to particular types of evaluation tasks on particular systems rather than being defined purely as the evaluation of what have been popularly termed "AI agents." This understanding of evaluation disambiguates the system being evaluated from the evaluation tasks; unlike model-only evaluations, evaluations of compound AI systems are highly dependent on the system architecture and on the components of the system other than the model.

Full definitions and a formal mathematical framework for agentic evaluations are presented in Appendix A.

3. Methodological Challenges in Agentic Evaluation of AI Systems

This section surveys and analyzes methodological challenges in agentic evaluations. We provide five high-level classes of challenges, i.e., challenges in concept development, system design, environmental interaction, scoring, and analysis & documentation. Our scope is limited only to those problems that are specific to or particularly salient in the context of agentic evaluations (e.g., scoring agentic evaluation outputs is difficult due to complex rubrics)—and we avoid discussing problems with evaluations in general (e.g., the cost of human grading is high). We focus on methodological choices that have implications for the validity, reliability, replicability, and cost/efficiency of agentic evaluations.

A selection of challenges are included in the main text of this workshop paper, with a fuller (work-in-progress) list presented in Appendix B.

3.1. Challenges in Concept Development

Concept development refers to the refinement of the underlying idea of interest to be measured by an evaluation, as well as the systematization of that idea into a well-scoped definition and related metrics for measurement (Adcock & Collier, 2001; Wallach et al., 2025).

How can performance in agentic evaluations be predicted, and when are non-agentic evaluations sufficient to measure capabilities or risks? Due to extended task horizons and difficulties in scoring, agentic evaluations can be challenging and resource-intensive to design, implement, and execute. Prior work has explored correlations (Schaeffer et al., 2025) and predictions (Zhou et al., 2025) between different question-answer benchmarks , but no research has examined whether these benchmarks or other metrics can accurately predict performance on agentic evaluations. Understanding predictors of performance in agentic evaluations may help evaluators identify the extent to which, relative to non-agentic evaluations, agentic evaluations offer additional information about compound AI system capabilities—and in which circumstances this additional information justifies their use.

How can evaluation concepts efficiently account for large state and action spaces with diverse solution pathways? Agentic evaluation tasks are characterized in part by the large state and action space in which an AI system is located. The proliferation of pathways to both task completion and failure—due to the functionally unconstrained nature of the state and action spaces—makes it important to measure concepts that account for the entire process of attempting the agentic evaluation task (see Pencharz et al., 2024; Yadav et al., 2019). One possible solution is to measure multiple concepts in a single evaluation, which (Wang et al., 2024) has suggested in the fairness context. Designing evaluation suites that measures the correct concepts of interest requires additional research.

In safety evaluations, what (or to what extent do) proxy tasks accurately reflect real-world risk? In the agentic evaluation context, evaluators may not be able to directly test for task completion of, e.g., a system's ability to create dangerous biological agents due to legal and ethical concerns. Evaluators will need to rely on proxy tasks to measure dangerous capabilities, and more clarity is needed on the extent to which such proxies signal capabilities and deployment behavior. Proxies in contexts that do not have direct human analogues, such as AI control (Greenblatt et al., 2024; Phuong et al., 2024), may be particularly diffi-

[2] This definition is roughly in line with prior work such as Zaharia et al., 2024; Lin et al., 2024a; Chen et al., 2024. Most systems termed AI agents would qualify as compound AI systems, though the converse is not true. Note also that our definition is based on systems' architecture rather than on systems' (intended) functions or capabilities, which is frequently the case with definitions of "AI agent" (e.g., the ability to independently complete tasks OpenAI, 2025; Shavit et al., 2023; Zittrain, 2024; Kolt, 2025).

cult to develop due to lack of precedents from other fields. In addition, evaluations may need to be behind closed doors or on isolated systems, which may affect the external validity of proxy tasks as well as present challenges around verification.

3.2. Challenges in System Design

Challenges in system design are those that relate to the selection of, design of, and interactions between the components of compound AI systems.

How should scaffolds be chosen? Scaffolds are code built to connect a model to external tools or other model instances; there is currently little consensus on scaffold best practices, and there are a wide variety of scaffolding frameworks available both commercially and through open source providers (OpenAI, 2025; LangChain). Wijk et al. (2024), for instance, conducted agentic evaluation across systems using two different scaffolding frameworks. Many evaluators create custom scaffolds for their evaluations, leading to few large-scale comparisons of different scaffolds. Because choice of scaffold can affect evaluation results, scaffolding is important to reliable and robust agentic evaluations.

How does task performance scale with improved scaffolds? An open question is to what extent improved scaffolds will lead to improved performance. METR (2025a) compared a simple agent scaffold and an "elicited agent" with a propose-evaluate CoT cycle and found that model performance improved with the "elicited agent." There is a challenge in evaluating whether improved performance due to improved scaffolding represents a genuine improvement in model capabilities, represents overfitting to a particular set of tasks, or represents lowering the difficulty of those tasks.

3.3. Challenges in Environmental Interactions

Challenges in environmental interactions relate to the design and selection of tools and (test) environments in which agentic evaluations occur.

How can the evaluation environment mimic the affordances available in real-world settings? If the environment provides fewer affordances to the model than it will have during real-world use, evaluation results may differ from what systems can actually achieve in deployment settings. For example, evaluations that do not provide internet access or that use weak scaffolds may systematically underestimate the performance of the system when capabilities are fully elicited by end users (see Appendix B.2.1).

How does performance generalize to tasks with novel toolsets? While early evaluations of tool use tested general-purpose LLMs on bespoke tool sets, the general question is whether evaluation results for one toolset generalize to those for other toolsets—e.g., Qin et al. (2023) evaluated a model on a different set of tools than it was trained on. This question may grow in importance as the third-party tool provider ecosystem grows, e.g., Anthropic's open-source Model Context Protocol (MCP) standard for agentic tool use (Anthropic, 2024b).

How do evaluators avoid trivial solutions? If the evaluation environment provides *too many* affordances during training, it may inflate scores beyond what we should expect during real use. For example, in PAPERBENCH, the authors ensure that systems are never allowed to view the original codebases of papers they have been tasked to replicate (Starace et al., 2025). More research is needed to develop practices for monitoring and preventing trivial solutions.

3.4. Challenges in Scoring

Challenges in scoring relate to the assignment of grades or scores to evaluation outputs, which could include both environmental state changes or system-generated artifacts.

How can robust scoring rubrics be efficiently developed across a spectrum of task complexity? Some evaluation contexts lend themselves to simple verification, while other contexts demand substantially more complex scoring. In the latter contexts, evaluators have created detailed rubrics via consultation with domain experts in what is often a resource-intensive process (Pencharz et al., 2024; Starace et al., 2025). In the context of non-agentic evaluations with no gold-standard labels, these rubrics have been shown to significantly affect final scores (Pathak et al., 2025; Hashemi et al., 2024; Fan et al., 2024). This result is likely to hold in the agentic evaluation context. Given the crucial role that rubrics play in evaluation scoring and subsequent results, additional research is needed to make rubric creation more valid and efficient.

How can evaluation results assign partial credit and minimize mode effects from scoring scales? Mode effects are errors in measurement due to particular measurement instruments (e.g., the order of questions, or whether a survey respondent answered a questionnaire via phone or online) rather than due to true differences in the underlying metrics of interest (Wei et al., 2025b). The scale on which model outputs are scored can create significant mode effects. Historically, many evaluations adopted a binary pass-fail scoring scale, which did not permit assignment of partial credit. Most recently, Phuong et al. (2024) and Shah et al. (2025) have suggested defining task milestones that would permit capturing performance improvements at higher fidelity. Additional research is needed as to how to best set these milestones and to reduce mode effects from different scoring choices.

How can autograders be validated and evaluated? Validation and evaluation of of AI graders has been a topic of increasing interest (Shankar et al., 2024; Guerdan et al., 2025). The complexity of agentic evaluation makes validation more important: for instance, Pencharz et al. (2024) developed a detailed step-by-step rubric for use with AI graders and found a systematic failure in AI graders to be biased towards leniency despite specific requirements in the grading rubric. The increased state/action space and possible subtasks may make validation a particularly thorny challenge in agentic evaluations.

3.5. Challenges in Analysis & Documentation

Challenges in analysis relate to the interpretation of evaluation results. Researchers have raised significant concerns with respect to the lack of statistical rigor in foundation model evaluations (e.g., Biderman & Scheirer, 2020), and best practices for analysis remain undetermined (e.g., Miller, 2024; Bowyer et al., 2025).

How can evaluators account and correct for diverse sources of statistical uncertainty? The complex and long-horizon nature of agentic evaluation tasks introduces many additional sources of bias and uncertainty as compared to traditional MCQA evaluation settings. These sources of uncertainty include: sampling error from the task space, sampling error from small sample sizes, sampling error from the grading process, construct error in the operationalization of measurement constructs, and systematic error from space of evaluation vs. performance tasks. No existing literature has attempted to systematically catalog and measure the effects of different sources of error, nor is it obvious how evaluators can implement corrections either in study design or post-hoc.

How can human baselines account for differences in modes of interaction between humans and AI systems? Preliminary evidence has suggested that AI systems are subject to mode effects, and that these mode effects are different from those experienced by humans (Tjuatja et al., 2024). These effects may affect the reliability and reproducibility of agentic evaluations, in addition to the validity of comparisons to human baselines; additional research is needed to quantify, control for, and correct for these effects.

How can evaluators measure and control for cost in humans and in AI systems, and what are the proper conversion rates between human and AI results? Cost metrics in agentic evaluations such as dollar cost or time are crucial for standardizing measurements across evaluation results (Kapoor et al., 2024), as well as for making comparisons between human baselines and AI results (Wei et al., 2025b). For instance, Rein et al. (2025) and Wijk et al. (2024) compare performance between human and AI systems on software task given the same length of time. However, the validity and units of comparisons have not been rigorously explored. These comparisons are of significant interest to downstream deployers/users, as well as to economic policymakers, and more work will be needed to build agentic evaluations that can predict AI's labor impacts.

4. Agentic Evaluations and AI Governance

The rise and importance of agentic evaluations has significant implications for AI governance frameworks, especially frameworks created at a time when MCQA evaluations were standard. Some implications are discussed below.

Evaluation validity & risk assurance. Because of the large state and action spaces of environmental tasks, the environmental validity of agentic evaluations may be particularly difficult to ensure. Standard benchmark approaches may not adequately reflect risk or capability, so it will be important to set evidentiary standards and required assurance levels for policies such as red lines or risk thresholds.

More complex supply chains. Agentic evaluations examine not just the underlying foundation model but also other system components such as scaffolding and tools, making the supply chain for compound AI systems more complex. Although recent work has examined downstream model developers (Williams et al., 2025), AI governance researchers may also wish to discuss the extent to which developers of non-model system components as well as service providers at the application layer should be regulated (see also Chan et al., 2025). For instance, developers may increase marginal risk by releasing, e.g., improved scaffolds or tools rather than improved models themselves.

High cost of agentic evaluations. Agentic evaluation tasks are complex and costly to develop, and benchmarks such as MMLU with nearly 16000 questions may not be feasible in agentic settings. The high cost is due to not just the length of the tasks but also to the high context and domain expertise required to develop meaningful agentic evaluation. As a result, it may not always be feasible to expect private actors to be willing to bear the costs or develop the required expertise to develop high-quality agentic evaluations. There may be more demand for publicly funded or built evaluation organizations, datasets, tools, and testbeds.

Competitive dynamics of non-model components. Analyses of competition in AI have centered on foundation model or compute providers (e.g., (Hua & Belfield, 2021; Narechania & Sitaraman, 2024)). Model and service providers may compete around non-model components, e.g., by creating platforms or industry-led standards such as A2A (Surapaneni et al., 2025) or MCP (Anthropic, 2024b). This shift has been underexplored in the literature, and it may have implications for safety (e.g., race dynamics) and antitrust.

Acknowledgements

The authors would like to thank Joshua Joseph, Jonathan Zittrain, and three anonymous reviewers from the ICML Technical AI Governance Workshop for feedback on earlier drafts of this work.

References

Adcock, R. and Collier, D. Measurement Validity: A Shared Standard for Qualitative and Quantitative Research. *American Political Science Review*, 95 (3):529–546, September 2001. ISSN 0003-0554, 1537-5943. doi: 10.1017/S0003055401003100. URL https://www.cambridge.org/core/journals/american-political-science-review/article/measurement-validity-a-shared-standard-for-qualitative-and-quantitative-research/91C7A9800DB26A76EBBABC5889A50C8B.

Ahuja, K., Sclar, M., and Tsvetkov, Y. Finding Flawed Fictions: Evaluating Complex Reasoning in Language Models via Plot Hole Detection, April 2025. URL http://arxiv.org/abs/2504.11900. arXiv:2504.11900 [cs].

Anthropic. Developing a computer use model, October 2024a. URL https://www.anthropic.com/news/developing-computer-use.

Anthropic. Model Context Protocol, 2024b. URL https://modelcontextprotocol.io/introduction.

Anthropic. Claude's extended thinking, February 2025. URL https://www.anthropic.com/news/visible-extended-thinking.

Baker, B., Huizinga, J., Gao, L., Dou, Z., Guan, M. Y., Madry, A., Zaremba, W., Pachocki, J., and Farhi, D. Monitoring Reasoning Models for Misbehavior and the Risks of Promoting Obfuscation, March 2025. URL http://arxiv.org/abs/2503.11926. arXiv:2503.11926 [cs].

Bengio, Y., Mindermann, S., Privitera, D., Besiroglu, T., Bommasani, R., Casper, S., Choi, Y., Fox, P., Garfinkel, B., Goldfarb, D., Heidari, H., Ho, A., Kapoor, S., Khalatbari, L., Longpre, S., Manning, S., Mavroudis, V., Mazeika, M., Michael, J., Newman, J., Ng, K. Y., Okolo, C. T., Raji, D., Sastry, G., Seger, E., Skeadas, T., South, T., Strubell, E., Tramèr, F., Velasco, L., Wheeler, N., Acemoglu, D., Adekanmbi, O., Dalrymple, D., Dietterich, T. G., Felten, E. W., Fung, P., Gourinchas, P.-O., Heintz, F., Hinton, G., Jennings, N., Krause, A., Leavy, S., Liang, P., Ludermir, T., Marda, V., Margetts, H., McDermid, J., Munga, J., Narayanan, A., Nelson, A., Neppel, C., Oh, A., Ramchurn, G., Russell, S., Schaake, M., Schölkopf, B., Song, D., Soto, A., Tiedrich, L., Varoquaux, G., Yao, A., Zhang, Y.-Q., Albalawi, F., Alserkal, M., Ajala, O., Avrin, G., Busch, C., Carvalho, A. C. P. d. L. F. d., Fox, B., Gill, A. S., Hatip, A. H., Heikkilä, J., Jolly, G., Katzir, Z., Kitano, H., Krüger, A., Johnson, C., Khan, S. M., Lee, K. M., Ligot, D. V., Molchanovskyi, O., Monti, A., Mwamanzi, N., Nemer, M., Oliver, N., Portillo, J. R. L., Ravindran, B., Rivera, R. P., Riza, H., Rugege, C., Seoighe, C., Sheehan, J., Sheikh, H., Wong, D., and Zeng, Y. The International Scientific Report on the Safety of Advanced AI. Technical Report DSIT 2025/001, UK Department for Science, Innovation and Technology, January 2025. URL http://arxiv.org/abs/2501.17805. arXiv:2501.17805 [cs].

Benton, J., Wagner, M., Christiansen, E., Anil, C., Perez, E., Srivastav, J., Durmus, E., Ganguli, D., Kravec, S., Shlegeris, B., Kaplan, J., Karnofsky, H., Hubinger, E., Grosse, R., Bowman, S. R., and Duvenaud, D. Sabotage Evaluations for Frontier Models, October 2024. URL http://arxiv.org/abs/2410.21514. arXiv:2410.21514 [cs].

Bestgen, Y. Exact Expected Average Precision of the Random Baseline for System Evaluation. *Prague Bulletin of Mathematical Linguistics*, 105:131–138, 2015. doi: 10.1515/pralin-2015-0007. URL https://ufal.mff.cuni.cz/pbml/103/art-bestgen.pdf.

Biderman, S. and Scheirer, W. J. Pitfalls in Machine Learning Research: Reexamining the Development Cycle. In *Proceedings on "I Can't Believe It's Not Better!" at NeurIPS Workshops*, pp. 106–117. PMLR, February 2020. URL https://proceedings.mlr.press/v137/biderman20a.html. ISSN: 2640-3498.

Boiko, D. A., MacKnight, R., Kline, B., and Gomes, G. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023. doi: https://doi.org/10.1038/s41586-023-06792-0. URL https://www.nature.com/articles/s41586-023-06792-0.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., Arx, S. v., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis,

J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the Opportunities and Risks of Foundation Models, July 2022. URL http://arxiv.org/abs/2108.07258. arXiv:2108.07258 [cs].

Bowyer, S., Aitchison, L., and Ivanova, D. R. Position: Don't use the CLT in LLM evals with fewer than a few hundred datapoints, March 2025. URL http://arxiv.org/abs/2503.01747. arXiv:2503.01747 [cs].

Burden, J., Tešić, M., Pacchiardi, L., and Hernández-Orallo, J. Paradigms of AI Evaluation: Mapping Goals, Methodologies and Culture, February 2025. URL http://arxiv.org/abs/2502.15620. arXiv:2502.15620 [cs].

Cai, L., Choi, K., Hansen, M., and Harrell, L. Item Response Theory. *Annual Review of Statistics and its Application*, 3:297–321, June 2016. doi: https://doi.org/10.1146/annurev-statistics-041715-033702. URL https://www.annualreviews.org/content/journals/10.1146/annurev-statistics-041715-033702.

Chan, A., Salganik, R., Markelius, A., Pang, C., Rajkumar, N., Krasheninnikov, D., Langosco, L., He, Z., Duan, Y., Carroll, M., Lin, M., Mayhew, A., Collins, K., Molamohammadi, M., Burden, J., Zhao, W., Rismani, S., Voudouris, K., Bhatt, U., Weller, A., Krueger, D., and Maharaj, T. Harms from Increasingly Agentic Algorithmic Systems. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '23, pp. 651–666, New York, NY, USA, June 2023. Association for Computing Machinery. ISBN 979-8-4007-0192-4. doi: 10.1145/3593013.3594033. URL https://dl.acm.org/doi/10.1145/3593013.3594033.

Chan, A., Wei, K., Huang, S., Rajkumar, N., Perrier, E., Lazar, S., Hadfield, G. K., and Anderljung, M. Infrastructure for AI Agents, January 2025. URL http://arxiv.org/abs/2501.10114. arXiv:2501.10114 [cs].

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., and Xie, X. A Survey on Evaluation of Large Language Models. *ACM Trans. Intell. Syst. Technol.*, 15(3):39:1–39:45, March 2024. ISSN 2157-6904. doi: 10.1145/3641289. URL https://dl.acm.org/doi/10.1145/3641289.

Chen, C., Yao, B., Zou, R., Hua, W., Lyu, W., Ye, Y., Li, T. J.-J., and Wang, D. Towards a Design Guideline for RPA Evaluation: A Survey of Large Language Model-Based Role-Playing Agents, March 2025. URL http://arxiv.org/abs/2502.13012. arXiv:2502.13012 [cs].

Chen, L., Davis, J. Q., Hanin, B., Bailis, P., Stoica, I., Zaharia, M., and Zou, J. Are More LLM Calls All You Need? Towards the Scaling Properties of Compound AI Systems. In *Advances in Neural Information Processing Systems*, November 2024. URL https://openreview.net/forum?id=m5106RRLgx&noteId=9l2RswepcT.

Cheng, C.-A., Kolobov, A., Misra, D., Nie, A., and Swaminathan, A. LLF-Bench: Benchmark for Interactive Learning from Language Feedback. *arXiv*, December 2023. doi: https://doi.org/10.48550/arXiv.2312.06853. URL https://arxiv.org/abs/2312.06853.

Chiang, W.-L., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhu, B., Zhang, H., Jordan, M., Gonzalez, J. E., and Stoica, I. Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference. In *Proceedings of the 41st International Conference on Machine Learning*, June 2024. URL https://openreview.net/forum?id=3MW8GKNyzI.

DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. Technical report, DeepSeek-AI, January 2025. URL https://github.com/deepseek-ai/DeepSeek-R1/blob/main/DeepSeek_R1.pdf.

Dell'Acqua, F., McFowland III, E., Mollick, E. R., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., Krayer, L., Candelon, F., and Lakhani, K. R. Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and

Quality, September 2023. URL https://papers.ssrn.com/abstract=4573321.

Dell'Acqua, F., Ayoubi, C., Lifshitz, H., Sadun, R., Mollick, E. R., Mollick, L., Han, Y., Goldman, J., Nair, H., Taub, S., and Lakhani, K. R. The Cybernetic Teammate: A Field Experiment on Generative Ai Reshaping Teamwork and Expertise, April 2025. URL https://papers.ssrn.com/abstract=5207588.

Dyba, T., Moe, N., and Arisholm, E. Measuring software methodology usage: challenges of conceptualization and operationalization. In *2005 International Symposium on Empirical Software Engineering, 2005.*, pp. 11 pp.–, November 2005. doi: 10.1109/ISESE.2005.1541852. URL https://ieeexplore.ieee.org/abstract/document/1541852.

Emmerich, K., Bogacheva, N., Bockholt, M., and Wendel, V. Operationalization and Measurement of Evaluation Constructs. In Dörner, R., Göbel, S., Kickmeier-Rust, M., Masuch, M., and Zweig, K. (eds.), *Entertainment Computing and Serious Games: International GI-Dagstuhl Seminar 15283, Dagstuhl Castle, Germany, July 5-10, 2015, Revised Selected Papers*, pp. 306–331. Springer International Publishing, Cham, 2016. ISBN 978-3-319-46152-6. doi: 10.1007/978-3-319-46152-6_13. URL https://doi.org/10.1007/978-3-319-46152-6_13.

Fan, Z., Wang, W., W, X., and Zhang, D. SedarEval: Automated Evaluation using Self-Adaptive Rubrics. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 16916–16930, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.984. URL https://aclanthology.org/2024.findings-emnlp.984/.

Feuer, B., Goldblum, M., Datta, T., Nambiar, S., Besaleli, R., Dooley, S., Cembalest, M., and Dickerson, J. P. Style Outweighs Substance: Failure Modes of LLM Judges in Alignment Benchmarking. In *Proceedings of the 13th International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=MzHNftnAM1.

Fish, S., Gonczarowski, Y. A., and Shorrer, R. I. Algorithmic Collusion by Large Language Models. *arXiv*, March 2024. doi: https://doi.org/10.48550/arXiv.2404.00806. URL https://arxiv.org/abs/2404.00806.

Fish, S., Shephard, J., Li, M., Shorrer, R., and Gonczarowksi, Y. A. EconEvals: Benchmarks and Litmus Tests for LLM Agents in Unknown Environments. *arXiv*, March 2025. doi: https://doi.org/10.48550/arXiv.2503.18825. URL https://arxiv.org/abs/2503.18825.

Geng, J., Cai, F., Wang, Y., Koeppl, H., Nakov, P., and Gurevych, I. A Survey of Confidence Estimation and Calibration in Large Language Models, March 2024. URL http://arxiv.org/abs/2311.08298. arXiv:2311.08298 [cs].

Gligoric, K., Zrnic, T., Lee, C., Candes, E., and Jurafsky, D. Can Unconfident LLM Annotations Be Used for Confident Conclusions? In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3514–3533, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. URL https://aclanthology.org/2025.naacl-long.179/.

Goemans, A., Buhl, M. D., Schuett, J., Korbak, T., Wang, J., Hilton, B., and Irving, G. Safety case template for frontier AI: A cyber inability argument, November 2024. URL http://arxiv.org/abs/2411.08088. arXiv:2411.08088 [cs].

Greenblatt, R., Shlegeris, B., Sachan, K., and Roger, F. AI Control: Improving Safety Despite Intentional Subversion. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 16295–16336. PMLR, July 2024. URL https://proceedings.mlr.press/v235/greenblatt24a.html. ISSN: 2640-3498.

Guan, S., Xiong, H., Wang, J., Bian, J., Zhu, B., and Lou, J.-g. Evaluating LLM-based Agents for Multi-Turn Conversations: A Survey, March 2025. URL http://arxiv.org/abs/2503.22458. arXiv:2503.22458 [cs].

Guerdan, L., Barocas, S., Holstein, K., Wallach, H., Wu, Z. S., and Chouldechova, A. Validating LLM-as-a-Judge Systems in the Absence of Gold Labels, March 2025. URL http://arxiv.org/abs/2503.05965. arXiv:2503.05965 [cs].

Guth, S. and Sapsis, T. P. Machine learning predictors of extreme events occurring in complex dynamical systems. *Entropy*, 21(10):925, September 2019. doi: https://doi.org/10.3390/e21100925. URL https://www.mdpi.com/1099-4300/21/10/925.

Götting, J., Medeiros, P., Sanders, J. G., Li, N., Phan, L., Elabd, K., Justen, L., Hendrycks, D., and Donoughe, S. Virology Capabilities Test (VCT): A Multimodal Virology Q&A Benchmark, April 2025. URL http://arxiv.org/abs/2504.16137. arXiv:2504.16137 [cs].

Hammond, L., Chan, A., Clifton, J., Hoelscher-Obermaier, J., Khan, A., McLean, E., Smith, C., Barfuss, W., Foerster, J., Gavenčiak, T., Han, T. A., Hughes, E., Kovařík, V., Kulveit, J., Leibo, J. Z., Oesterheld, C., Witt, C. S. d., Shah, N., Wellman, M., Bova, P., Cimpeanu, T., Ezell, C., Feuillade-Montixi, Q., Franklin, M., Kran, E., Krawczuk, I., Lamparth, M., Lauffer, N., Meinke, A., Motwani, S., Reuel, A., Conitzer, V., Dennis, M., Gabriel, I., Gleave, A., Hadfield, G., Haghtalab, N., Kasirzadeh, A., Krier, S., Larson, K., Lehman, J., Parkes, D. C., Piliouras, G., and Rahwan, I. Multi-Agent Risks from Advanced AI, February 2025. URL http://arxiv.org/abs/2502.14143. arXiv:2502.14143 [cs].

Hashemi, H., Eisner, J., Rosset, C., Van Durme, B., and Kedzie, C. LLM-Rubric: A Multidimensional, Calibrated Approach to Automated Evaluation of Natural Language Texts. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13806–13834, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.745. URL https://aclanthology.org/2024.acl-long.745/.

He, Q., Zeng, J., Huang, W., Chen, L., Xiao, J., He, Q., Zhou, X., Liang, J., and Xiao, Y. Can large language models understand real-world complex instructions? In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, volume 38 of *AAAI'24/IAAI'24/EAAI'24*, pp. 18188–18196. AAAI Press, February 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i16.29777. URL https://doi.org/10.1609/aaai.v38i16.29777.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring Massive Multitask Language Understanding. *arXiv*, September 2020. doi: https://doi.org/10.48550/arXiv.2009.03300. URL https://arxiv.org/abs/2009.03300.

Hsieh, C.-P., Sun, S., Kriman, S., Acharya, S., Rekesh, D., Jia, F., and Ginsburg, B. RULER: What's the Real Context Size of Your Long-Context Language Models? In *Proceedings of the 1st Conference on Language Modeling*, August 2024. URL https://openreview.net/forum?id=kIoBbc76Sy#discussion.

Hua, S.-S. and Belfield, H. AI & Antitrust: Reconciling Tensions Between Competition Law and Cooperative AI Development. *Yale Journal of Law & Technology*, 23:415, 2021. URL https://yjolt.org/ai-antitrust-reconciling-tensions-between-competition-law-and-cooperative-ai-development.

Huang, Q., Vora, J., Liang, P., and Leskovec, J. MLAgentBench: evaluating language agents on machine learning experimentation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *ICML'24*, pp. 20271–20309, Vienna, Austria, July 2024. JMLR.org.

Huang, Y., Shi, J., Li, Y., Fan, C., Wu, S., Zhang, Q., Liu, Y., Zhou, P., Wan, Y., Gong, N. Z., and Sun, L. MetaTool Benchmark for Large Language Models: Deciding Whether to Use Tools and Which to Use. In *Proceedings of the the 12th International Conference on Learning Representations*, October 2023. URL https://openreview.net/forum?id=R0c2qtalgG.

Jimenez, C. E., Yang, J., Wettig, A., Yao, S., Pei, K., Press, O., and Narasimhan, K. SWE-bench: Can Language Models Resolve Real-World GitHub Issues?, November 2024. URL http://arxiv.org/abs/2310.06770. arXiv:2310.06770 [cs].

Kamradt, G. LLMTest_needleinahaystack, 2024. URL https://github.com/gkamradt/LLMTest_NeedleInAHaystack.

Kapoor, S., Stroebl, B., Siegel, Z. S., Nadgir, N., and Narayanan, A. AI Agents That Matter, July 2024. URL http://arxiv.org/abs/2407.01502. arXiv:2407.01502 [cs].

Katsoulakis, M. A. and Plechac, P. Information-theoretic tools for parametrized coarse-graining of non-equilibrium extended systems. *The Journal of Chemical Physics*, 139(7):074115–14, August 2013. doi: https://doi.org/10.1063/1.4818534. URL https://pubs.aip.org/aip/jcp/article-abstract/139/7/074115/73364/Information-theoretic-tools-for-parametrized?redirectedFrom=fulltext.

Kolt, N. Governing AI Agents, February 2025. URL https://papers.ssrn.com/abstract=4772956.

Koo, R., Lee, M., Raheja, V., Park, J. I., Kim, Z. M., and Kang, D. Benchmarking Cognitive Biases in Large Language Models as Evaluators. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 517–545, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.29. URL https://aclanthology.org/2024.findings-acl.29/.

Kwa, T., West, B., Becker, J., Deng, A., Garcia, K., Hasin, M., Jawhar, S., Kinniment, M., Rush, N., Arx, S. V., Bloom, R., Broadley, T., Du, H., Goodrich, B., Jurkovic, N., Miles, L. H., Nix, S., Lin, T., Parikh, N., Rein, D., Sato, L. J. K., Wijk, H., Ziegler, D. M., Barnes, E., and Chan, L. Measuring AI Ability to Complete Long Tasks, March 2025. URL http://arxiv.org/abs/2503.14499. arXiv:2503.14499 [cs].

Laine, R., Chughtai, B., Betley, J., Hariharan, K., Scheurer, J., Balesni, M., Hobbhahn, M., Meinke, A., and Evans, O. Me, Myself, and AI: The Situational Awareness Dataset (SAD) for LLMs, July 2024. URL http://arxiv.org/abs/2407.04694. arXiv:2407.04694 [cs].

LangChain. LangChain. URL https://www.langchain.com/.

Laurent, J. M., Janizek, J. D., Ruzo, M., Hinks, M. M., Hammerling, M. J., Narayanan, S., Ponnapati, M., White, A. D., and Rodriques, S. G. LAB-Bench: Measuring Capabilities of Language Models for Biology Research, July 2024. URL http://arxiv.org/abs/2407.10362. arXiv:2407.10362 [cs].

Leng, Q., Portes, J., Havens, S., Zaharia, M., and Carbin, M. Long Context RAG Performance of Large Language Models, November 2024. URL http://arxiv.org/abs/2411.03538. arXiv:2411.03538 [cs].

Li, H., Chen, J., Yang, J., Ai, Q., Wei, J., Liu, Y., Lin, K., Wu, Y., Yuan, G., Hu, Y., Wan, W., Liu, Y., and Huang, M. LegalAgentBench: Evaluating LLM Agents in Legal Domain, December 2024a. URL https://arxiv.org/abs/2412.17259v1.

Li, L., Wang, Y., Zhao, H., Kong, S., Teng, Y., Li, C., and Wang, Y. Reflection-Bench: probing AI intelligence with reflection. *arXiv*, October 2024b. doi: https://doi.org/10.48550/arXiv.2410.16270. URL https://arxiv.org/abs/2410.16270.

Light, J., Xing, S., Liu, Y., Chen, W., Cai, M., Chen, X., Wang, G., Cheng, W., Yue, Y., and Hu, Z. PIANIST: Learning Partially Observable World Models with LLMs for Multi-Agent Decision Making, November 2024. URL http://arxiv.org/abs/2411.15998. arXiv:2411.15998 [cs].

Lin, M., Sheng, J., Zhao, A., Wang, S., Yue, Y., Wu, Y., Liu, H., Liu, J., Huang, G., and Liu, Y.-J. LLM-based Optimization of Compound AI Systems: A Survey, October 2024a. URL http://arxiv.org/abs/2410.16392. arXiv:2410.16392 [cs].

Lin, Z., Trivedi, S., and Sun, J. Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models. *Transactions on Machine Learning Research*, February 2024b. ISSN 2835-8856. URL https://openreview.net/forum?id=DWkJCSxKU5.

Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., Zhang, S., Deng, X., Zeng, A., Du, Z., Zhang, C., Shen, S., Zhang, T., Su, Y., Sun, H., Huang, M., Dong, Y., and Tang, J. AgentBench: Evaluating LLMs as Agents. In *Proceedings of the 12th International Conference on Learning Representations*, October 2023. URL https://openreview.net/forum?id=zAdUB0aCTQ.

Liu, X., Chen, T., Da, L., Chen, C., Lin, Z., and Wei, H. Uncertainty Quantification and Confidence Calibration in Large Language Models: A Survey, March 2025. URL http://arxiv.org/abs/2503.15850. arXiv:2503.15850 [cs].

Lu, J., Holleis, T., Zhang, Y., Aumayer, B., Nan, F., Bai, H., Ma, S., Ma, S., Li, M., Yin, G., Wang, Z., and Pang, R. ToolSandbox: A Stateful, Conversational, Interactive Evaluation Benchmark for LLM Tool Use Capabilities. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1160–1183, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. URL https://aclanthology.org/2025.findings-naacl.65/.

Luo, J., Zhang, W., Yuan, Y., Zhao, Y., Yang, J., Gu, Y., Wu, B., Chen, B., Qiao, Z., Long, Q., Tu, R., Luo, X., Ju, W., Xiao, Z., Wang, Y., Xiao, M., Liu, C., Yuan, J., Zhang, S., Jin, Y., Zhang, F., Wu, X., Zhao, H., Tao, D., Yu, P. S., and Zhang, M. Large Language Model Agent: A Survey on Methodology, Applications and Challenges, March 2025. URL http://arxiv.org/abs/2503.21460. arXiv:2503.21460 [cs].

Malinin, A. and Gales, M. Uncertainty Estimation in Autoregressive Structured Prediction. In *Proceedings of the 8th International Conference on Learning Representations*, October 2020. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

Manvi, R., Singh, A., and Ermon, S. Adaptive Inference-Time Compute: LLMs Can Predict if They Can Do Better, Even Mid-Generation. *arXiv*, October 2024. doi: https://doi.org/10.48550/arXiv.2410.02725. URL https://arxiv.org/abs/2410.02725.

Meng, K., Huang, V., Steinhardt, J., and Schwettmann, S. Introducing Docent: A system for analyzing and intervening on agent behavior, March 2025. URL https://transluce.org/introducing-docent.

METR. An update on our preliminary evaluations of Claude 3.5 Sonnet and o1. Technical report, METR, January 2025a.

METR. Measuring Automated Kernel Engineering. Technical report, METR, February 2025b. URL https://metr.org/blog/2025-02-14-measuring-automated-kernel-engineering/.

METR. Details about METR's preliminary evaluation of OpenAI's o3 and o4-mini. Technical report, METR, April 2025c. URL https://metr.github.io/autonomy-evals-guide/openai-o3-report/.

Mialon, G., Fourrier, C., Wolf, T., LeCun, Y., and Scialom, T. GAIA: a benchmark for General AI Assistants. In *Proceedings of the 12th International Conference on Learning Representations*, October 2023. URL https://openreview.net/forum?id=fibxvahvs3.

Miller, E. Adding Error Bars to Evals: A Statistical Approach to Language Model Evaluations, November 2024. URL http://arxiv.org/abs/2411.00640. arXiv:2411.00640.

Mundhenk, M., Goldsmith, J., Lusena, C., and Allender, E. Complexity of finite-horizon Markov decision process problems. *Journal of the ACM*, 37(4):681–720, July 2000. doi: https://doi.org/10.1145/347476.347480. URL https://dl.acm.org/doi/10.1145/347476.347480.

Narechania, T. N. and Sitaraman, G. An Antimonopoly Approach to Governing Artificial Intelligence. *Yale Law & Policy Review*, 43(1), 2024. URL https://yalelawandpolicy.org/antimonopoly-approach-governing-artificial-intelligence.

OpenAI. Introducing ChatGPT Pro, December 2024. URL https://openai.com/index/introducing-chatgpt-pro/.

OpenAI. A practical guide to building agents, April 2025. URL https://cdn.openai.com/business-guides-and-resources/a-practical-guide-to-building-agents.pdf.

Pan, J., Shar, R., Pfau, J., Talwalkar, A., He, H., and Chen, V. When Benchmarks Talk: Re-Evaluating Code LLMs with Interactive Feedback. *arXiv*, February 2025. doi: https://doi.org/10.48550/arXiv.2502.18413. URL https://arxiv.org/abs/2502.18413.

Panickssery, A., Bowman, S. R., and Feng, S. LLM Evaluators Recognize and Favor Their Own Generations. In *Advances in Neural Information Processing Systems*, November 2024. URL https://openreview.net/forum?id=4NJBV6Wp0h.

Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 979-8-4007-0132-0. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763. event-place: San Francisco, CA, USA.

Pathak, A., Gandhi, R., Uttam, V., Devansh, Nakka, Y., Jindal, A. R., Ghosh, P., Ramamoorthy, A., Verma, S., Mittal, A., Ased, A., Khatri, C., Challa, J. S., and Kumar, D. Rubric Is All You Need: Enhancing LLM-based Code Evaluation With Question-Specific Rubrics, March 2025. URL http://arxiv.org/abs/2503.23989. arXiv:2503.23989 [cs].

Patil, S. G., Zhang, T., Wang, X., and Gonzalez, J. E. Gorilla: Large Language Model Connected with Massive APIs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 126544–126565, December 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e4c61f578ff07830f5c37378dd3ecb0d-Abstract-Conference.html.

Pencharz, J., Flesch, T., and Sandbrink, J. Long-Form Tasks: A Methodology for Evaluating Scientific Assistants. Technical report, AI Security Institute, December 2024. URL https://www.aisi.gov.uk/work/long-form-tasks.

Phan, L., Gatti, A., Han, Z., Li, N., Hu, J., Zhang, H., Zhang, C. B. C., Shaaban, M., Ling, J., Shi, S., Choi, M., Agrawal, A., Chopra, A., Khoja, A., Kim, R., Ren, R., Hausenloy, J., Zhang, O., Mazeika, M., Dodonov, D., Nguyen, T., Lee, J., Anderson, D., Doroshenko, M., Stokes, A. C., Mahmood, M., Pokutnyi, O., Iskra, O., Wang, J. P., Levin, J.-C., Kazakov, M., Feng, F., Feng, S. Y., Zhao, H., Yu, M., Gangal, V., Zou, C., Wang, Z., Popov, S., Gerbicz, R., Galgon, G., Schmitt, J., Yeadon, W., Lee, Y., Sauers, S., Sanchez, A., Giska, F., Roth, M., Riis, S., Utpala, S., Burns, N., Goshu, G. M., Naiya, M. M., Agu, C., Giboney, Z., Cheatom, A., Fournier-Facio, F., Crowson, S.-J., Finke, L., Cheng, Z., Zampese, J., Hoerr, R. G., Nandor, M., Park, H., Gehrunger, T., Cai, J., McCarty, B., Garretson, A. C., Taylor, E., Sileo, D., Ren, Q., Qazi, U., Li, L., Nam, J., Wydallis, J. B., Arkhipov, P., Shi, J. W. L., Bacho, A., Willcocks, C. G., Cao, H., Motwani, S., Santos, E. d. O., Veith, J., Vendrow, E., Cojoc, D., Zenitani, K., Robinson, J., Tang, L., Li, Y., Vendrow, J., Fraga, N. W., Kuchkin, V., Maksimov, A. P., Marion, P., Efremov, D., Lynch, J., Liang, K., Mikov, A., Gritsevskiy, A., Guillod, J., Demir, G., Martinez, D., Pageler, B., Zhou, K., Soori, S., Press, O., Tang, H., Rissone, P., Green, S. R., Brüssel, L., Twayana, M., Dieuleveut, A., Imperial, J. M., Prabhu, A., Yang, J., Crispino, N., Rao, A., Zvonkine, D., Loiseau, G., Kalinin, M., Lukas, M., Manolescu, C., Stambaugh, N., Mishra, S., Hogg, T., Bosio, C., Coppola, B. P., Salazar, J., Jin, J., Sayous, R., Ivanov, S., Schwaller, P., Senthilkuma, S., Bran, A. M., Algaba, A., Houte, K. V. d., Sypt, L. V. D., Verbeken, B., Noever, D., Kopylov, A., Myklebust, B., Li, B., Schut, L., Zheltonozhskii, E., Yuan, Q., Lim, D., Stanley, R., Yang, T., Maar, J., Wykowski, J., Oller, M., Sahu, A., Ardito, C. G., Hu, Y., Kamdoum, A. G. K., Jin, A., Vilchis, T. G., Zu, Y., Lackner, M., Koppel, J., Sun, G., Antonenko, D. S., Chern, S., Zhao, B., Arsene, P., Cavanagh, J. M., Li, D., Shen, J., Crisostomi, D., Zhang, W., Dehghan, A., Ivanov, S., Perrella, D., Kaparov, N., Zang, A., Sucholutsky, I., Kharlamova, A., Orel, D., Poritski, V., Ben-David, S., Berger, Z., Whitfill, P., Foster, M., Munro, D., Ho, L., Sivarajan, S., Hava, D. B., Kuchkin, A., Holmes, D., Rodriguez-Romero, A., Sommerhage, F., Zhang, A., Moat, R., Schneider, K., Kazibwe, Z., Clarke, D., Kim, D. H., Dias, F. M., Fish, S., Elser, V., Kreiman, T., Vilchis, V. E. G., Klose, I., Anantheswaran, U., Zweiger, A., Rawal, K., Li, J., Nguyen, J., Daans, N., Heidinger, H., Radionov, M., Rozhoň, V., Ginis, V., Stump, C., Cohen, N., Poświata, R., Tkadlec, J., Goldfarb, A., Wang, C., Padlewski, P., Barzowski, S., Montgomery, K., Stendall, R., Tucker-Foltz, J., Stade, J., Rogers, T. R., Goertzen, T., Grabb, D., Shukla, A., Givré, A., Ambay, J. A., Sen, A., Aziz, M. F., Inlow, M. H., He, H., Zhang, L., Kaddar, Y., Ängquist, I., Chen, Y., Wang, H. K., Ramakrishnan, K., Thornley, E., Terpin, A., Schoelkopf, H., Zheng, E., Carmi, A., Brown, E. D. L., Zhu, K., Bartolo, M., Wheeler, R., Stehberger, M., Bradshaw, P., Heimonen, J. P., Sridhar, K., Akov, I., Sandlin, J., Makarychev, Y., Tam, J., Hoang, H., Cunningham, D. M., Goryachev, V., Patramanis, D., Krause, M., Redenti, A., Aldous, D., Lai, J., Coleman, S., Xu, J., Lee, S., Magoulas, I., Zhao, S., Tang, N., Cohen, M. K., Paradise, O., Kirchner, J. H., Ovchynnikov, M., Matos, J. O., Shenoy, A., Wang, M., Nie, Y., Sztyber-Betley, A., Faraboschi, P., Riblet, R., Crozier, J., Halasyamani, S., Verma, S., Joshi, P., Meril, E., Ma, Z., Andréoletti, J., Singhal, R., Platnick, J., Nevirkovets, V., Basler, L., Ivanov, A., Khoury, S., Gustafsson, N., Piccardo, M., Mostaghimi, H., Chen, Q., Singh, V., Khánh, T. Q., Rosu, P., Szlyk, H., Brown, Z., Narayan, H., Menezes, A., Roberts, J., Alley, W., Sun, K., Patel, A., Lamparth, M., Reuel, A., Xin, L., Xu, H., Loader, J., Martin, F., Wang, Z., Achilleos, A., Preu, T., Korbak, T., Bosio, I., Kazemi, F., Chen, Z., Bálint, B., Lo, E. J. Y., Wang, J., Nunes, M. I. S., Milbauer, J., Bari, M. S., Wang, Z., Ansarinejad, B., Sun, Y., Durand, S., Elgnainy, H., Douville, G., Tordera, D., Balabanian, G., Wolff, H., Kvistad, L., Milliron, H., Sakor, A., Eron, M., O, A. F. D., Shah, S., Zhou, X., Kamalov, F., Abdoli, S., Santens, T., Barkan, S., Tee, A., Zhang, R., Tomasiello, A., Luca, G. B. D., Looi, S.-Z., Le, V.-K., Kolt, N., Pan, J., Rodman, E., Drori, J., Fossum, C. J., Muennighoff, N., Jagota, M., Pradeep, R., Fan, H., Eicher, J., Chen, M., Thaman, K., Merrill, W., Firsching, M., Harris, C., Ciobâcă, S., Gross, J., Pandey, R., Gusev, I., Jones, A., Agnihotri, S., Zhelnov, P., Mofayezi, M., Piperski, A., Zhang, D. K., Dobarskyi, K., Leventov, R., Soroko, I., Duersch, J., Taamazyan, V., Ho, A., Ma, W., Held, W., Xian, R., Zebaze, A. R., Mohamed, M., Leser, J. N., Yuan, M. X., Yacar, L., Lengler, J., Olszewska, K., Fratta, C. D., Oliveira, E., Jackson, J. W., Zou, A., Chidambaram, M., Manik, T., Haffenden, H., Stander, D., Dasouqi, A., Shen, A., Golshani, B., Stap, D., Kretov, E., Uzhou, M., Zhidkovskaya, A. B., Winter, N., Rodriguez, M. O., Lauff, R., Wehr, D., Tang, C., Hossain, Z., Phillips, S., Samuele, F., Ekström, F., Hammon, A., Patel, O., Farhidi, F., Medley, G., Mohammadzadeh, F., Peñaflor, M., Kassahun, H., Friedrich, A., Perez, R. H., Pyda, D., Sakal, T., Dhamane, O., Mirabadi, A. K., Hallman, E., Okutsu, K., Battaglia, M., Maghsoudimehrabani, M., Amit, A., Hulbert, D., Pereira, R., Weber, S., Handoko, Peristyy, A., Malina, S., Mehkary, M., Aly, R., Reidegeld, F., Dick, A.-K., Friday, C., Singh, M., Shapourian, H., Kim, W., Costa, M., Gurdogan, H., Kumar, H., Ceconello, C., Zhuang, C., Park, H., Carroll, M., Tawfeek, A. R., Steinerberger, S., Aggarwal, D., Kirchhof, M., Dai, L., Kim, E., Ferret, J., Shah, J., Wang, Y., Yan, M., Burdzy, K., Zhang, L., Franca, A., Pham, D. T., Loh, K. Y., Robinson, J., Jackson, A., Giordano, P., Petersen, P., Cosma, A., Colino, J., White, C., Votava, J., Vinnikov, V., Delaney, E., Spelda, P., Stritecky, V., Shahid, S. M., Mourrat, J.-C., Vetoshkin, L., Sponselee, K., Bacho, R., Yong, Z.-X., Rosa, F. d. l., Cho, N., Li, X., Malod, G., Weller, O., Albani, G., Lang, L., Laurendeau, J., Kazakov, D., Adesanya, F., Portier, J., Hollom, L., Souza, V., Zhou, Y. A., Degorre, J., Yalın, Y., Obikoya, G. D., Rai, Bigi, F., Boscá, M. C., Shumar, O., Bacho, K., Recchia, G., Popescu, M., Shulga, N., Tanwie, N. M., Lux, T. C. H., Rank, B., Ni, C., Brooks, M., Yakimchyk, A., Huanxu, Liu, Cavalleri, S., Häggström, O., Verkama, E., Newbould, J., Gundlach, H., Brito-Santana, L., Amaro, B., Vajipey, V., Grover, R., Wang, T., Kratish, Y., Li, W.-D., Gopi, S., Caciolai, A., Witt, C. S. d., Hernández-Cámara, P., Rodolà, E., Robins, J., Williamson, D., Cheng, V., Raynor, B., Qi, H., Segev, B., Fan, J., Martinson, S., Wang, E. Y., Hausknecht, K., Brenner, M. P., Mao, M., Demian, C., Kassani, P., Zhang, X., Avagian, D., Scipio, E. J., Ragoler, A., Tan, J., Sims, B., Plecnik, R., Kirtland, A., Bodur, O. F., Shinde, D. P., Labrador, Y. C. L., Adoul, Z., Zekry, M., Karakoc, A., Santos, T. C. B., Shamseldeen, S., Karim, L., Liakhovitskaia, A., Resman, N., Farina, N., Gonzalez, J. C., Maayan, G., Anderson, E., Pena, R. D. O., Kelley, E., Mariji, H., Pouriamanesh, R., Wu, W., Finocchio, R., Alarab, I., Cole, J., Ferreira, D., Johnson, B., Safdari, M., Dai, L., Arthornthurasuk, S., McAlister, I. C., Moyano, A. J., Pronin, A., Fan, J., Ramirez-Trinidad, A., Malysheva, Y., Pottmaier, D., Taheri, O., Stepanic, S., Perry, S., Askew, L., Rodríguez, R. A. H., Minissi, A. M. R., Lorena, R., Iyer, K., Fasiludeen, A. A., Clark, R., Ducey, J., Piza, M., Somrak, M., Vergo, E., Qin, J., Borbás, B., Chu, E., Lindsey, J., Jallon, A., McInnis, I. M. J., Chen, E., Semler, A., Gloor, L., Shah, T., Carauleanu, M., Lauer, P., Huy, T. , Shahrtash, H., Duc, E., Lewark, L., Brown, A., Albanie, S., Weber, B., Vaz, W. S., Clavier, P., Fan, Y., Silva, G. P. R. e., Long, Lian, Abramovitch, M., Jiang, X., Mendoza, S., Islam, M., Gonzalez, J., Mavroudis, V., Xu, J., Kumar, P., Goswami, L. P., Bugas, D., Heydari, N., Jeanplong, F., Jansen, T., Pinto, A., Apronti, A., Galal, A., Ze-An, N., Singh, A., Jiang, T., Xavier, J. o. A., Agarwal, K. P., Berkani, M., Zhang, G., Du, Z., Junior, B. A. d. O., Malishev, D., Remy, N., Hartman, T. D., Tarver, T., Mensah, S., Loume, G. A., Morak, W., Habibi, F., Hoback, S., Cai, W., Gimenez, J., Montecillo, R. G., Łucki, J., Campbell, R., Sharma, A., Meer, K., Gul, S., Gonzalez, D. E., Alapont, X., Hoover, A., Chhablani, G., Vargus, F., Agarwal, A., Jiang, Y., Patil, D., Outevsky, D., Scaria, K. J., Maheshwari, R., Dendane, A., Shukla, P., Cartwright, A., Bogdanov, S., Mündler, N., Möller, S., Arnaboldi, L., Thaman, K., Siddiqi, M. R., Saxena, P., Gupta, H., Fruhauff, T., Sherman, G., Vincze, M., Usawasutsakorn, S., Ler, D., Radhakrishnan, A., Enyekwe, I., Salauddin, S. M., Muzhen, J., Maksapetyan, A., Rossbach, V., Harjadi, C., Bahaloohoreh, M., Sparrow, C., Sidhu, J., Ali, S., Bian, S., Lai, J., Singer, E., Uro, J. L., Bateman, G., Sayed, M., Menshawy, A., Duclosel, D., Bezzi, D., Jain, Y., Aaron, A., Tiryakioglu, M., Siddh, S., Krenek, K., Shah, I. A., Jin, J., Creighton, S., Peskoff, D., EL-Wasif, Z., V, R. P., Richmond, M., McGowan, J., Patwardhan, T., Sun, H.-Y., Sun, T., Zubić, N., Sala, S., Ebert, S., Kaddour, J., Schottdorf, M., Wang, D., Petruzella, G., Meiburg, A., Medved, T., ElSheikh, A., Hebbar, S. A., Vaquero, L., Yang, X., Poulos, J., Zouhar, V., Bogdanik, S., Zhang, M., Sanz-Ros, J., Anugraha, D., Dai, Y., Nhu, A. N., Wang, X., Demircali, A. A., Jia, Z., Zhou, Y., Wu, J., He, M., Chandok, N., Sinha, A., Luo, G., Le, L., Noyé, M., Perełkiewicz, M., Pantidis, I., Qi, T., Purohit, S. S., Parcalabescu, L., Nguyen, T.-H., Winata, G. I., Ponti, E. M., Li, H., Dhole, K., Park, J., Abbondanza, D., Wang, Y., Nayak, A., Caetano, D. M., Wong, A. A. W. L., Rio-Chanona, M. d., Kondor, D., Francois, P., Chalstrey, E., Zsambok, J., Hoyer, D., Reddish, J., Hauser, J., Rodrigo-Ginés, F.-J., Datta, S., Shepherd, M., Kamphuis, T., Zhang, Q., Kim, H., Sun, R., Yao, J., Dernoncourt, F., Krishna, S., Rismanchian, S., Pu, B., Pinto, F., Wang, Y., Shridhar, K., Overholt, K. J., Briia, G., Nguyen, H., David, Bartomeu, S., Pang, T. C., Wecker, A., Xiong, Y., Li, F., Huber, L. S., Jaeger, J., Maddalena, R. D., Lù, X. H., Zhang, Y., Beger, C., Kon, P. T. J., Li, S., Sanker, V., Yin, M., Liang, Y., Zhang, X., Agrawal, A., Yifei, L. S., Zhang, Z., Cai, M., Sonmez, Y., Cozianu, C., Li, C., Slen, A., Yu, S., Park, H. K., Sarti, G., Briański, M., Stolfo, A., Nguyen, T. A., Zhang, M., Perlitz, Y., Hernandez-Orallo, J., Li, R., Shabani, A., Juefei-Xu, F., Dhingra, S., Zohar, O., Nguyen, M. C., Pondaven, A., Yilmaz, A., Zhao, X., Jin, C., Jiang, M., Todoran, S., Han, X., Kreuer, J., Rabern, B., Plassart, A., Maggetti, M., Yap, L., Geirhos, R., Kean, J., Wang, D., Mollaei, S., Sun, C., Yin, Y., Wang, S., Li, R., Chang, Y., Wei, A., Bizeul, A., Wang, X., Arrais, A. O., Mukherjee, K., Chamorro-Padial, J., Liu, J., Qu, X., Guan, J., Bouyamourn, A., Wu, S., Plomecka, M., Chen, J., Tang, M., Deng, J., Subramanian, S., Xi, H., Chen, H., Zhang, W., Ren, Y., Tu, H., Kim, S., Chen, Y., Marjanović, S. V., Ha, J., Luczyna, G., Ma, J. J., Shen, Z., Song, D., Zhang, C. E., Wang, Z., Gendron, G., Xiao, Y., Smucker, L., Weng, E., Lee, K. H., Ye, Z., Ermon, S., Lopez-Miguel, I. D., Knights, T., Gitter, A., Park, N., Wei, B., Chen, H., Pai, K., Elkhanany, A., Lin, H., Siedler, P. D., Fang, J., Mishra, R., Zsolnai-Fehér, K., Jiang, X., Khan, S., Yuan, J., Jain, R. K., Lin, X., Peterson, M., Wang, Z., Malusare, A., Tang, M., Gupta, I., Fosin, I., Kang, T., Dworakowska, B., Matsumoto, K., Zheng, G., Sewuster, G., Villanueva, J. P., Rannev, I., Chernyavsky, I., Chen, J., Banik, D., Racz, B., Dong, W., Wang, J., Bashmal, L., Gonçalves, D. V., Hu, W., Bar, K., Bohdal, O., Patlan, A. S., Dhuliawala, S., Geirhos, C., Wist, J., Kansal, Y., Chen, B., Tire, K., Yücel, A. T., Christof, B., Singla, V., Song, Z., Chen, S., Ge, J., Ponkshe, K., Park, I., Shi, T., Ma, M. Q., Mak, J., Lai, S., Moulin, A., Cheng, Z., Zhu, Z., Zhang, Z., Patil, V., Jha, K., Men, Q., Wu, J., Zhang, T., Vieira, B. H., Aji, A. F., Chung, J.-W., Mahfoud, M., Hoang, H. T., Sperzel, M., Hao, W., Meding, K., Xu, S., Kostakos, V., Manini, D., Liu, Y., Toukmaji, C., Paek, J., Yu, E., Demircali, A. E., Sun, Z., Dewerpe, I., Qin, H., Pflugfelder, R., Bailey, J., Morris, J., Heilala, V., Rosset, S., Yu, Z., Chen, P. E., Yeo, W., Jain, E., Yang, R., Chigurupati, S., Chernyavsky, J., Reddy, S. P., Venugopalan, S., Batra, H., Park, C. F., Tran, H., Maximiano, G., Zhang, G., Liang, Y., Shiyu, H., Xu, R., Pan, R., Suresh, S., Liu, Z., Gulati, S., Zhang, S., Turchin, P., Bartlett, C. W., Scotese, C. R., Cao, P. M., Nattanmai, A., McKellips, G., Cheraku, A., Suhail, A., Luo, E., Deng, M., Luo, J., Zhang, A., Jindel, K., Paek, J., Halevy, K., Baranov, A., Liu, M., Avadhanam, A., Zhang, D., Cheng, V., Ma, B., Fu, E., Do, L., Lass, J., Yang, H., Sunkari, S., Bharath, V., Ai, V., Leung, J., Agrawal, R., Zhou, A., Chen, K., Kalpathi, T., Xu, Z., Wang, G., Xiao, T., Maung, E., Lee, S., Yang, R., Yue, R., Zhao, B., Yoon, J., Sun, S., Singh, A., Luo, E., Peng, C., Osbey, T., Wang, T., Echeazu, D., Yang, H., Wu, T., Patel, S., Kulkarni, V., Sundarapandiyan, V., Zhang, A., Le, A., Nasim, Z., Yalam, S., Kasamsetty, R., Samal, S., Yang, H., Sun, D., Shah, N., Saha, A., Zhang, A., Nguyen, L., Nagumalli, L., Wang, K., Zhou, A., Wu, A., Luo, J., Telluri, A., Yue, S., Wang, A., and Hendrycks, D. Humanity's Last Exam, April 2025. URL http://arxiv.org/abs/2501.14249. arXiv:2501.14249 [cs].

Phuong, M., Aitchison, M., Catt, E., Cogan, S., Kaskasoli, A., Krakovna, V., Lindner, D., Rahtz, M., Assael, Y., Hodkinson, S., Howard, H., Lieberum, T., Kumar, R., Raad, M. A., Webson, A., Ho, L., Lin, S., Farquhar, S., Hutter, M., Deletang, G., Ruoss, A., El-Sayed, S., Brown, S., Dragan, A., Shah, R., Dafoe, A., and Shevlane, T. Evaluating Frontier Models for Dangerous Capabilities, April 2024. URL http://arxiv.org/abs/2403.13793. arXiv:2403.13793 [cs].

Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y., Lin, Y., Cong, X., Tang, X., Qian, B., Zhao, S., Hong, L., Tian, R., Xie, R., Zhou, J., Gerstein, M., Li, D., Liu, Z., and Sun, M. ToolLLM: Facilitating Large Language Models to Master 16000_ Real-world APIs. In *ICLR 2024*, 2023. URL https://openreview.net/pdf?id=dHng2O0Jjr.

Raji, D., Denton, E., Bender, E. M., Hanna, A., and Paullada, A. AI and the Everything in the Whole Wide World Benchmark. In *Advances in Neural Information Processing Systems*, volume 1, December 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/084b6fbb10729ed4da8c3d3f5a3ae7c9-Abstract-round2.html.

Rein, D., Hou, B. L., Stickland, A. C., Petty, J., Pang, R. Y., Dirani, J., Michael, J., and Bowman, S. R. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. In *Proceedings of the 1st Conference on Language Modeling*, August 2024. URL https://openreview.net/forum?id=Ti67584b98#discussion.

Rein, D., Becker, J., Deng, A., Nix, S., Canal, C., O'Connel, D., Arnott, P., Bloom, R., Broadley, T., Garcia, K., Goodrich, B., Hasin, M., Jawhar, S., Kinniment, M., Kwa, T., Lajko, A., Rush, N., Sato, L. J. K., Arx, S. V., West, B., Chan, L., and Barnes, E. HCAST: Human-Calibrated Autonomy Software Tasks, March 2025. URL http://arxiv.org/abs/2503.17354. arXiv:2503.17354 [cs].

Reuel, A., Hardy, A., Smith, C., Lamparth, M., Hardy, M., and Kochenderfer, M. BetterBench: Assessing AI Benchmarks, Uncovering Issues, and Establishing Best Practices. In *Advances in Neural Information Processing Systems*, November 2024. URL https://openreview.net/forum?id=hcOq2buakM#discussion.

Schaeffer, R., Koura, P. S., Tang, B., Subramanian, R., Singh, A. K., Mihaylov, T., Bhargava, P., Madaan, L., Chatterji, N. S., Goswami, V., Edunov, S., Hupkes, D., Koyejo, S., and Narang, S. Correlating and Predicting Human Evaluations of Language Models from Natural Language Processing Benchmarks, February 2025. URL http://arxiv.org/abs/2502.18339. arXiv:2502.18339 [cs].

Schwarcz, D., Manning, S., Barry, P., Cleveland, D. R., Prescott, J., and Rich, B. AI-Powered Lawyering: AI Reasoning Models, Retrieval Augmented Generation, and the Future of Legal Practice. Technical Report 25-16, University of Minnesota Law School, March 2025. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5162111.

Shah, R., Irpan, A., Turner, A. M., Wang, A., Conmy, A., Lindner, D., Brown-Cohen, J., Ho, L., Nanda, N., Popa, R. A., Jain, R., Greig, R., Albanie, S., Emmons, S., Farquhar, S., Krier, S., Rajamanoharan, S., Bridgers, S., Ijitoye, T., Everitt, T., Krakovna, V., Varma, V., Mikulik, V., Kenton, Z., Orr, D., Legg, S., Goodman, N., Dafoe, A., Flynn, F., and Dragan, A. An Approach to Technical AGI Safety and Security, April 2025. URL http://arxiv.org/abs/2504.01849. arXiv:2504.01849 [cs].

Shankar, S., Zamfirescu-Pereira, J., Hartmann, B., Parameswaran, A., and Arawjo, I. Who Validates the Validators? Aligning LLM-Assisted Evaluation of LLM Outputs with Human Preferences. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, UIST '24, pp. 1–14, New York, NY, USA, October 2024. Association for Computing Machinery. ISBN 979-8-4007-0628-8. doi: 10.1145/3654777.3676450. URL https://dl.acm.org/doi/10.1145/3654777.3676450.

Shavit, Y., Agarwal, S., Brundage, M., Adler, S., O'Keefe, C., Campbell, R., Lee, T., Mishkin, P., Eloundou, T., Hickey, A., Slama, K., Ahmad, L., McMillan, P., Beutel, A., Passos, A., and Robinson, D. G. Practices for Governing Agentic AI Systems, December 2023. URL https://cdn.openai.com/papers/practices-for-governing-agentic-ai-systems.pdf.

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K. R., and Yao, S. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, November 2023. URL https://openreview.net/forum?id=vAElhFcKW6.

Spaan, M. T. Partially Observable Markov Decision Processes. In *Adaptation, Learning, and Optimization*, volume 12 of *ALO*, pp. 387–414. Springer, Berlin, Heidelberg, 2012. ISBN 1867-4542. URL https://link.springer.com/chapter/10.1007/978-3-642-27645-3_12.

Starace, G., Jaffe, O., Sherburn, D., Aung, J., Chan, J. S., Maksin, L., Dias, R., Mays, E., Kinsella, B., Thompson, W., Heidecke, J., Glaese, A., and Patwardhan, T. PaperBench: Evaluating AI's Ability to Replicate AI Research, April 2025. URL http://arxiv.org/abs/2504.01848. arXiv:2504.01848 [cs].

Surapaneni, R., Jha, M., Vakoc, M., and Segal, T. Announcing the Agent2Agent Protocol (A2A), April 2025. URL https://developers.googleblog.com/en/a2a-a-new-era-of-agent-interoperability/.

Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., and Manning, C. Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL https://aclanthology.org/2023.emnlp-main.330/.

Tjuatja, L., Chen, V., Wu, T., Talwalkwar, A., and Neubig, G. Do LLMs Exhibit Human-like Response Biases? A Case Study in Survey Design. *Transactions of the Association for Computational Linguistics*, 12:1011–1026, September 2024. ISSN 2307-387X. doi: 10.1162/tacl_a_00685. URL https://doi.org/10.1162/tacl_a_00685.

Toner, H., Bansemer, J., Crichton, K., Burtell, M., Woodside, T., Lior, A., Lohn, A., Acharya, A., Cibralic, B., Painter, C., O'Keefe, C., Gabriel, I., Fisher, K., Ramakrishnan, K., Jackson, K., Kolt, N., Crootof, R., and Chatterjee, S. Through the Chat Window and Into the Real World: Preparing for AI Agents. Analysis, Center for Security and Emerging Technology, October 2024. URL https://cset.georgetown.edu/publication/through-the-chat-window-and-into-the-real-world-preparing-for-ai-agents/.

UK AI Security Institute. Early lessons from evaluating frontier AI systems | AISI Work, October 2024. URL https://www.aisi.gov.uk/work/early-lessons-from-evaluating-frontier-ai-systems.

Vries, C. M. D., Geva, S., and Trotman, A. Document Clustering Evaluation: Divergence from a Random Baseline, August 2012. URL http://arxiv.org/abs/1208.5654. arXiv:1208.5654 [cs].

Wagner, N., Desmond, M., Nair, R., Ashktorab, Z., Daly, E. M., Pan, Q., Cooper, M. S., Johnson, J. M., and Geyer, W. Black-box Uncertainty Quantification Method for LLM-as-a-Judge, October 2024. URL https://arxiv.org/abs/2410.11594v1.

Wallach, H., Desai, M., Cooper, A. F., Wang, A., Atalla, C., Barocas, S., Blodgett, S. L., Chouldechova, A., Corvi, E., Dow, P. A., Garcia-Gathright, J., Olteanu, A., Pangakis, N., Reed, S., Shen, E., Vann, D., Vaughan, J. W., Vogel, M., Washington, H., and Jacobs, A. Z. Position: Evaluating Generative AI Systems is a Social Science Measurement Challenge, February 2025. URL https://arxiv.org/abs/2502.00561.

Wang, A., Hertzmann, A., and Russakovsky, O. Benchmark suites instead of leaderboards for evaluating AI fairness. *Patterns*, 5(11), November 2024. ISSN 2666-3899. doi: 10.1016/j.patter.2024.101080. URL https://www.cell.com/patterns/abstract/S2666-3899(24)00239-3. Publisher: Elsevier.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, NIPS '22, pp. 24824–24837, Red Hook, NY, USA, November 2022. Curran Associates Inc. ISBN 978-1-7138-7108-8.

Wei, J., Sun, Z., Papay, S., McKinney, S., Han, J., Fulford, I., Chung, H. W., Passos, A. T., Fedus, W., and Glaese, A. BrowseComp: A Simple Yet Challenging Benchmark for Browsing Agents, April 2025a. URL http://arxiv.org/abs/2504.12516. arXiv:2504.12516 [cs].

Wei, K., Paskov, P., Dev, S., Byun, M. J., Reuel, A., Roberts-Gaal, X., Calcott, R., Coxon, E., and Deshpande, C. Position: Model Evaluations Need Rigorous and Transparent Human Baselines. In *Proceedings of Machine Learning Research (Forthcoming)*, 2025b. URL https://openreview.net/forum?id=VbG9sIsn4F&referrer=%5Bthe%20profile%20of%20Kevin%20Wei%5D(%2Fprofile%3Fid%3D~Kevin_Wei1).

Wijk, H., Lin, T., Becker, J., Jawhar, S., Parikh, N., Broadley, T., Chan, L., Chen, M., Clymer, J., Dhyani, J., Ericheva, E., Garcia, K., Goodrich, B., Jurkovic, N., Kinniment, M., Lajko, A., Nix, S., Sato, L., Saunders, W., Taran, M., West, B., and Barnes, E. RE-Bench: Evaluating frontier AI R&D capabilities of language model agents against human experts, November 2024. URL http://arxiv.org/abs/2411.15114. arXiv:2411.15114 [cs].

Williams, S., Schuett, J., and Anderljung, M. On Regulating Downstream AI Developers, March 2025. URL http://arxiv.org/abs/2503.11922. arXiv:2503.11922 [cs].

Xander Davies [@alxndrdavies]. When we were developing our agent misuse dataset, we noticed instances of models seeming to realize our tasks were fake. We're sharing some examples and we'd be excited for more research into how synthetic tasks can distort eval results! [thread emoji] 1/N https://t.co/8wvGGRz7zd, February 2025. URL https://x.com/alxndrdavies/status/1890418558339022932.

Xie, T., Zhang, D., Chen, J., Li, X., Zhao, S., Cao, R., Hua, T. J., Cheng, Z., Shin, D., Lei, F., Liu, Y., Xu, Y., Zhou, S., Savarese, S., Xiong, C., Zhong, V., and Yu, T. OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments, May 2024. URL http://arxiv.org/abs/2404.07972. arXiv:2404.07972 [cs].

Yadav, D., Jain, R., Agrawal, H., Chattopadhyay, P., Singh, T., Jain, A., Singh, S. B., Lee, S., and Batra, D. EvalAI: Towards Better Evaluation Systems for AI Agents, February 2019. URL http://arxiv.org/abs/1902.03570. arXiv:1902.03570 [cs].

Yang, Y., Chai, H., Song, Y., Qi, S., Wen, M., Li, N., Liao, J., Hu, H., Lin, J., Chang, G., Liu, W., Wen, Y., Yu, Y., and Zhang, W. A Survey of AI Agent Protocols, April 2025. URL http://arxiv.org/abs/2504.16736. arXiv:2504.16736 [cs].

Yao, B., Chen, G., Zou, R., Lu, Y., Li, J., Zhang, S., Sang, Y., Liu, S., Hendler, J., and Wang, D. More Samples or More Prompts? Exploring Effective Few-Shot In-Context Learning for LLMs with In-Context Sampling. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 1772–1790, Mexico City, June 2024. Association for Computational Linguistics. doi: https://doi.org/10.18653/v1/2024.findings-naacl.115. URL https://aclanthology.org/2024.findings-naacl.115/.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., and Cao, Y. ReAct: Synergizing Reasoning and Acting in Language Models. In *Proceedings of the 11th International Conference on Learning Representations*, September 2022. URL https://openreview.net/forum?id=WE_vluYUL-X.

Yauney, G. and Mimno, D. Stronger Random Baselines for In-Context Learning. In *Proceedings of the 1st Conference on Language Modeling*, August 2024. URL https://openreview.net/forum?id=TRxQMpLUfD#discussion.

Ye, J., Wang, Y., Huang, Y., Chen, D., Zhang, Q., Moniz, N., Gao, T., Geyer, W., Huang, C., Chen, P.-Y., Chawla, N. V., and Zhang, X. Justice or Prejudice? Quantifying Biases in LLM-as-a-Judge. In *Proceedings of the 13th International Conference on Learning Representations*, October 2024. URL https://openreview.net/forum?id=3GTtZFiajM.

Yehudai, A., Eden, L., Li, A., Uziel, G., Zhao, Y., Bar-Haim, R., Cohan, A., and Shmueli-Scheuer, M. Survey on Evaluation of LLM-based Agents, March 2025. URL http://arxiv.org/abs/2503.16416. arXiv:2503.16416 [cs].

You, J., Liu, M., Prabhumoye, S., Patwary, M., Shoeybi, M., and Catanzaro, B. LLM-Evolve: Evaluation for LLM's Evolving Capability on Benchmarks. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 16937–16942, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: https://doi.org/10.18653/v1/2024.emnlp-main.940. URL https://aclanthology.org/2024.emnlp-main.940/.

Zaharia, M., Khattab, O., Chen, L., Davis, J. Q., Miller, H., Potts, C., Zou, J., Carbin, M., Frankle, J., Rao, N., and Ghodsi, A. The Shift from Models to Compound AI Systems, February 2024. URL http://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/.

Zeng, Z., Yu, J., Gao, T., Meng, Y., Goyal, T., and Chen, D. Evaluating Large Language Models at Evaluating Instruction Following. In *Proceedings of the 12th International Conference on Learning Representations*, October 2023. URL https://openreview.net/forum?id=tr0KidwPLc.

Zhang, Y., Chen, J., Wang, J., Liu, Y., Yang, C., Shi, C., Zhu, X., Lin, Z., Wan, H., Yang, Y., Sakai, T., Feng, T., and Yamana, H. ToolBeHonest: A Multi-level Hallucination Diagnostic Benchmark for Tool-Augmented Large Language Models. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11388–11422, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.637. URL https://aclanthology.org/2024.emnlp-main.637/.

Zhong, L., Du, Z., Zhang, X., Hi, H., and Tang, J. ComplexFuncBench: Exploring Multi-Step and Constrained Function Calling under Long-Context Scenario, January 2025. URL https://arxiv.org/abs/2501.10132.

Zhou, L., Pacchiardi, L., Martínez-Plumed, F., Collins, K. M., Moros-Daval, Y., Zhang, S., Zhao, Q., Huang, Y., Sun, L., Prunty, J. E., Li, Z., Sánchez-García, P., Chen, K. J., Casares, P. A. M., Zu, J., Burden, J., Mehrbakhsh, B., Stillwell, D., Cebrian, M., Wang, J., Henderson, P., Wu, S. T., Kyllonen, P. C., Cheke, L., Xie, X., and Hernández-Orallo, J. General Scales Unlock AI Evaluation with Explanatory and Predictive Power, March 2025. URL http://arxiv.org/abs/2503.06378. arXiv:2503.06378 [cs].

Zittrain, J. L. We Need to Control AI Agents Now, July 2024. URL https://www.theatlantic.com/technology/archive/2024/07/ai-agents-safety-risks/678864/. Section: Technology.

A. Background

A.1. Defining Compound AI Systems

We are interested in evaluations of compound AI systems based on foundation models, or general-purpose AI models trained on diverse data (Bommasani et al., 2022). In this context, "compound AI systems" are systems that consist of at least one component in addition to a foundation model instance (Zaharia et al., 2024; Lin et al., 2024a; Chen et al., 2024). These components could include:

- Scaffolding:[3] code built to connect a model to external tools or other model instances. Emerging standards for scaffolding include Anthropic's Model Context Protocol (Anthropic, 2024b) and Google's A2A (Surapaneni et al., 2025).
- Tools: functions or instruments that enable a model to interact with an external environment. Examples of tools include APIs or command line interfaces.
- Model instance(s): systems could include multiple foundation model instances. For example, a compound AI system could consist of a base model (to generate outputs) as well as a smaller monitor model (to check the outputs of the base model) (Baker et al., 2025). Compound AI systems composed of multiple interacting model instances are termed multi-agent systems (Hammond et al., 2025).
- Advanced prompting techniques, such as chain-of-thought reasoning.
- Data sources such as those used in retrieval augmented generation (RAG).

This definition has multiple advantages for the purposes of discussions on agentic evaluations—in particular, although most if not all AI agents would qualify as compound AI systems, not all compound AI systems are AI agents. Compound AI systems are characterized by system architectures and components, whereas definitions of "AI agent" frequently focus on systems' (intended) functions or capabilities (e.g., the ability to independently complete tasks (OpenAI, 2025; Shavit et al., 2023; Zittrain, 2024; Kolt, 2025)). Centering our discussion on compound AI systems allows us to focus on different parts of the system and how they affect task performance. It also avoids definitional debates around the AI agent terminology and prevents conflation of the term "AI agents" with philosophical or legal agency (Chan et al., 2023).

Our definition refocuses the unit of evaluation on the model instance, *plus additional components that enable system capabilities*. In contrast, many existing evaluations view scaffolding as a component of the evaluation setup rather than the model, the "toolbench model" of agentic evaluation where the foundation model is analogous to a craftsperson, while the evaluation provides 1) the tools to accomplish a task and 2) the metrics with which to measure task success. This shift in framing towards evaluation of the *entire* compound AI system rather than the foundation model alone better reflects real-world deployments of AI systems.

Our definition of compound AI systems explicitly excludes AI systems trained purely from reinforcement learning (RL) techniques, since these systems are normally not based on foundation models.[4] This exclusion is because foundation model evaluation—and by extension, evaluations of foundation model-based compound AI systems—have significant methodological differences and complexities as compared to traditional RL evaluation. The primary difference is that RL systems use evaluations as reward functions in a closed loop for optimization. In contrast, agentic evaluations are not intended for direct feedback within the training loop. Foundation model systems can, of course, also be applied to traditional RL settings such as games and robotics; we discuss some of these settings below, though work in this area is limited, and our discussion applies to these settings without loss of generality.

[3] One preliminary question is whether a scaffold is conceptually part of the model to be evaluated, or an intrinsic part of the task itself. In other words: do evaluators aim to benchmark the capabilities of 1) different base foundation models when plugged into a given scaffold and task, or 2) different scaffolded AI systems (which may use the same base model) on a given task? Historically, many evaluators have chosen the first option by building the scaffold into the evaluation suite. For example, AGENTBENCH (Liu et al., 2023) evaluation suite includes a separate scaffold for each of eight environments, and the authors evaluated 27 LLMs based on API calls. On the other hand, the second option allows evaluators can make use of existing tool affordances built into commercial AI systems. BROWSECOMP (Wei et al., 2025a), for instance, measured how well OpenAI models used their built-in web browsing tools to answer challenging research questions, while GAIA (Mialon et al., 2023) have a specific goal of evaluating the AI system as a unified whole. SWE-BENCH (Jimenez et al., 2024) also considers each agent's scaffold a part of its entry on the leaderboard.

[4] Compound AI systems based on foundation models trained partially with RL techniques remain in scope.

A.2. Defining Environmental Tasks

Like AI agents, compound AI systems can also complete *tasks*. We limit our discussion in this article to evaluations of compound AI systems' capabilities to complete environmental tasks, as defined below.

"Environmental tasks" are defined as activities that are:

- Environmental: environmental tasks require an AI system to engage with an external setting, usually by using tools.
- Under-specified: environmental tasks are not fully specified by the user; rather, environmental tasks require AI systems to engage in some level of independent reasoning or planning independent of the user (e.g., reasoning about how or which tools to use, or more generally making inferences not explicitly provided by the user).
- Multi-step: environmental tasks require AI systems to take a sequence of actions. These actions, or subtasks, can be separated by underlying models' interactions with other system components (e.g., writing and executing a query to a RAG database), with the user (e.g., asking for clarification), or with an external environment (e.g., making a tool call).[5]
- Scorable: environmental tasks are characterized by output quality or completion levels that can be systematically quantified or scored by a third-party (other than the user); i.e., success or failure of a task is not purely dependent on user opinion.[6]

In particular, environmental tasks can vary along any of the following dimensions:

- Single vs. multi-turn: a turn is one round of interaction between a user and an AI system. In some cases, a system may be able to complete tasks in a single turn, provided that the user includes sufficient context during the initial interaction or prompt; in other cases, a system may demand clarification from a user before proceeding with attempting task completion. A number of challenges arise in evaluation when these tasks become multi-turn, e.g., due to difficulties in quantifying the effects of subsequent turns on task completion or output quality. We will discuss many of these difficulties in Section B.[7]
- Output category: environmental tasks may produce a variety of outputs. For instance, the output could be an artifact provided to the user, or it could be a state changed produced in the external environment. Similarly, the output could be closed-form (e.g., a multiple-choice answer) or open-ended (e.g., free-form text).
- Scoring scale: environmental tasks could be graded on different scales. For instance, some tasks may be graded solely on task completion (binary scale), while others could be graded on output quality (e.g., Likert scale).
- Input/output modalities: environmental tasks could receive as input or produce as output data of any modality (e.g., text, video, audio, or other formats).

Some examples of tasks include software engineering, personal administration and time management, creation or grading of educational materials, drug development, or research assistance. Environmental tasks can often be economically valuable or be activities in which humans currently engage.

We exclude several related but distinct evaluation contexts from our discussion:

- Evaluations of non-task capabilities (e.g., benchmarks of model-only knowledge, information retrieval, or pure reasoning): benchmarks such as GPQA (Rein et al., 2024) or FLAWEDFICTIONS (Ahuja et al., 2025) do not use tools but rather test only properties internal to a model. On the other hand, tool-assisted information retrieval tasks such as (Wei et al., 2025a) are in-scope. Moreover, non-tool use evaluations such as GPQA could be converted to agentic evaluations of compound AI systems if completed with sufficient tools and scaffolding, though such evaluation items may not necessarily be a good fit for agentic evaluation (e.g., due to ease of saturation).

[5] One motivation for limiting our discussion primarily to multi-step tasks is that most tasks of sufficient complexity to be of economic interest are multi-step.

[6] Many tasks whose purpose is solely to satisfy the user may nevertheless be gradeable. For instance, environmental/tool-using conversational tasks could be assigned scores by an annotator per a fixed rubric.

[7] Because we are interested in evaluating *systems*, we exclude evaluations where the evaluation target is the product of a human and AI team (i.e., human uplift studies).

- Evaluations of non-task propensities in compound AI systems (e.g., alignment): although AI agents are particularly well-suited to completing tasks as specified above, “AI agent evaluation” frequently encompasses evaluation on non-task based propensities such as multiple-choice question-answer (MCQA)-based alignment or preferences Yehudai et al. 2025. We consider these evaluations out of scope this work; on the other hand, an alignment evaluation focused on revealed preferences (by studying AI systems’ actions, e.g., Fish et al., 2025) would be in-scope.

- Human uplift trials (e.g., (Dell’Acqua et al., 2025; 2023; Schwarcz et al., 2025): human uplift trials are not evaluations of *autonomous* model capabilities but rather evaluations of how much a AI system can assist a human in completing tasks.

- Non-environmental multi-turn tasks (e.g., Guan et al., 2025; Chen et al., 2025): tasks such as roleplaying or therapy chatbots that do not interact with external environments are out of scope, though environmental versions of the same tasks would be in scope.

A.3. Mathematical Description

This section briefly sets out a formalization of a compound AI system executing an environmental task. Section A.3.1 defines a task as partially observable Markov decision process. Section A.3.2 describes the mathematical framework for metric calculation for traditional LLM evaluations. Section A.3.3 modifies the notation from prior sections to accomodate agentic evaluations.

A.3.1. Description of a Task

The behavior of an AI system performing a task can be modeled as a decision-making process. The agent’s internal chain of thought is a scratchpad for reasoning to choose between a set of possible actions that affect the environment.

Decision processes under incomplete information may be formalized as partially observable Markov decision processes (POMDP) (Spaan, 2012). A POMDP is a tuple (S, A, Ω, T, O, R) in which S is a (finite) set of environmental states, A is a (finite) set of actions the agent can take, Ω is a (finite) set of observations available to the agent, T is a transition probability $T : S \times A \times S \to [0, 1]$ that gives the probability of a transition from one state to another given a choice of action by the agent, O is an observation probability $O : S \times A \times \Omega \to [0, 1]$ that gives the probability of the AI system receiving a particular observation given the environmental state and choice of action, and R is the reward function $R : S \times A \times S \to \mathbb{R}$. POMDPs may be defined for infinite time (simpler) or finite horizons (more complicated) (Mundhenk et al., 2000).

Traditional RL problems are formulated in a carefully limited environment, either in a simulation or a physical sandbox, where external influences are neglected or included stochastically. In these cases, the modeled environment S_{system} is a coarse-graining of the full environment S_{full}, and this coarse-graining introduces stochastic noise in the transition matrix T (Katsoulakis & Plechac, 2013). For many tasks considered here, the AI system cannot be completely sandboxed in a sterile computing because tool interacts with the world. This could be to a limited spatial extent, as in a robotics system that conducts chemistry experiments by interacting with lab hardware, or to a limited physical extent, as in a research system that accesses the open internet. (Boiko et al., 2023). One possibility is to partition the environmental state S into a local state S_{LC} which represents the state of local compute resources, and a nonlocal state S_N, which represents all other state variables which could affect the agent, and then coarse-graining over S_N.

A related question is what elements *internal* to the AI system to include in the system state S, because LLM based agents build an internal context during inference. Thus S_L can in turn be partitioned *internally* as the state of the context, S_C, and the state of the local compute resources exclusive of the context S_L. The internals of the decision making process is only relevant to the POMDP to the extent that it is captured by the agent’s policy, $\phi : O \to A$ the function that chooses actions based on observations about the environment.

The reward function R defines success: for state $s \in S$ and AI system action $a \in A$, $R(s, a)$ gives a value of zero if the task is not successfully completed, and a value of one if the task is successfully completed. For tasks in a virtual (compute) environment, the environment has a deterministic many-to-one mapping onto observations, but for tasks involving real world sensing this relationship is in general stochastic.

(Xie et al., 2024) simplify this definition of POMDPs by eschewing the stochastic behavior. In their formalism, the transition function is deterministic and is given by $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$. Further, they eliminate the observation probability measure O. Instead, they define the (deterministic) current observation $o_t \in \mathcal{O}$ (Ω in Spaan) as a complete screenshot of the desktop screen, an XML-format accessibility tree, and a terminal output. (Light et al., 2024) generalize this formulation to multi-agent decision-making systems that may include environmental stochasticity. (Liu et al., 2023) expanded the definition of POMDP to include $\mathcal{U}$, a separate task instruction space.

For RL problems, feedback from R is used to optimize policy ϕ, and the literature on POMDPs includes methods to calculate the optimal policy for a given problem statement. However, agentic evaluation is different from traditional RL in two ways. First, even with the appropriate simplifications (such as coarse-graining) above, the state and action spaces for agents attempting even everyday-type tasks are computationally intractable. Second, evaluators do not calculate the optimal policy in the first instance but rather evaluate particular policies defined implicitly by a particular compound AI systems.

In this evaluation posture, the reward function R is not evaluated along a trajectory. Instead, the evaluator extracts instances of R from the end of task evaluations, possibly after the AI system takes a special `task-complete` action.

A further simplification is the case of binary success, that is, $R(s) \in \{0, 1\}$. In the binary success case, an equivalent description is the set of states $s_y \in Y$ for which $R(s_y) = 1$ (or more formally, all (s, a) such that $R(s, a) = 1$ and $T(s, a) = s_y$). Environmental tasks do not necessarily have binary success: for instance, the reward may decrease as amount of time or number of steps taken by the AI system increases (preferring fast solutions to slow solution). In general, nonbinary success ($R(s) \in [0, 1]$) makes confusion matrix-based evaluation metrics more complicated (Guth & Sapsis, 2019).

A.3.2. Traditional LLM Evaluation

The simplest form of traditional evaluation of AI systems (such as LLMs) is the battery of multiple choice question. Evaluators first create a set of question-answer pairs $\mathcal{D} = \{x_i, y_i^*\}_{i \leq n}$ called the testing corpus. While in most cases $\mathcal{D}$ is constructed by expert human annotators for the purpose of evaluation, $\mathcal{D}$ may be considered a finite sample of some ideal distribution of evaluations question-answer pairs $\mathcal{D}^\infty$. In some cases, the testing corpus is subdivided into a representative public set and a held-out private set to avoid "training for the test," but here $\mathcal{D}$ refers only to the held-out set.

The questions presented to the AI system during the testing procedure may be either the entire question set $\mathcal{D}$, or some subset. Without loss of generality, let the $m \leq n$ presented questions have indices $1 \ldots m$. For a testing corpus with distinct subcorpora and subscores, this procedure may be applied to each subcorpus separately.

During the testing procedure, evaluators present the AI system with the question x_i and record the elicited response $\hat{y}(\omega_i)$, where ω_i represents the "random seed" used for model generations. For nondeterministic AI systems, such as LLMs with positive temperature $T > 0$, repeating the experiment with the same question may result in different elicited responses. For ease of presentation, the explicit dependence of $\hat{y}$ on ω_i will be generally left out for the remainder of this section.

After eliciting a set of responses $\hat{\mathcal{D}} = \{x_i, \hat{y}_i\}_{i \leq m}$, evaluators grade the AI system's responses to the presented questions to estimate the probability that the system will correctly answer a randomly drawn question from $\mathcal{D}$. That is, evaluations observe the empirical fraction

$$\hat{P} = \frac{1}{m} \sum_i^m \mathbb{I} \left[\hat{y}_i = y_i^* \right], \tag{1}$$

where $\hat{P}$ is the empirical performance and $\mathbb{I}$ is the indicator function. This empirical performance is an estimate for the true performance of the AI system, given by

$$P = \mathbb{E}_i \mathbb{E}_\omega \mathbb{I} \left[\hat{y}_i(\omega) = y_i^* \right], \tag{2}$$

where $\mathbb{E}_\omega$ is an expectation over the nondeterministic part of the AI system (and the explicit dependence of $\hat{y}_i$ on ω is temporarily restored) and $\mathbb{E}_i$ is an expectation over the set of all questions in the testing corpus $\mathcal{D}$ (which stands for an expectation over $\mathcal{D}^\infty$). Here, P is a true aleatoric probability, for the probability space $(\{(x, y)\},\ \sigma\{(x, y)\},\ \mathbb{P})$. Note that the probability an AI system answers a particular question correctly across many tries, $P[x = x_i] = \mathbb{E}_\omega \mathbb{I} \left[\hat{y}_i = y_i^* \right]$, is *not* that same as the probability that an AI system will answer any question from the set $\mathcal{D}$ correctly on a particular inference $P[\omega = \omega_i] = \mathbb{E}_i \mathbb{I} \left[\hat{y}_i = y_i^* \right]$.

This quantity P depends on a number of parameters. Directly, it depends on the choice of testing corpus $\mathcal{D}$. Similarly, it depends on the choice of AI system, which may include any fine tuning done to the system. For nondeterministic AI systems it depends on sampler settings, here represented with the temperature T by synecdoche.

Further, P depends on details of the evaluation prompt in two ways. First, it depends on the instruction formatting: whether the question x_i is presented "as-is" or wrapped in some kind of explanation. The field of prompt engineering examines how changing the format of the instructions can change the performance of an AI system. Second, P depends on whether the AI system is given access to example question-answer pairs. In zero-shot prompting the system is not given any examples or demonstrations, while in few-shot prompting the system is given some number of correct question-answer pairs.

Finally, the interpretation of P depends on theoretical work relating the testing corpus $\mathcal{D}$ to the actual constructs of interest, work that (Raji et al., 2021) describe as construct validity. In narrowly operationalized contexts, where the testing corpus closely resembles the expected application of the agentic system, the quantity P may directly represent success rate. In other cases, however, $\mathcal{D}$ is an instrument for estimating various latent capabilities based on Item Response Theory (IRT) (Cai et al., 2016). For instance, (Kwa et al., 2025) applied IRT to use task difficulty (as proxied through task completion time) for predictions of AI performance.

For some more involved evaluation analyses, the mean probability of correctly responding to a question drawn randomly from $\mathcal{D}$ may not capture all of the performance information. For instance, in a corpus of classification questions, the AI system may be penalized more for false positives than false negatives, or vice versa. Or, in the context of multiple choice questions, the evaluation may depend on which distractor the AI system selects. In general, P is just one summary statistic for the complete confusion matrix, and "accuracy" standing alone is an especially fraught statistic for problems with unbalanced base rates (Guth & Sapsis, 2019; Wagner et al., 2024).

During problem setup, the testing corpus $\mathcal{D}$ is assumed to be a representative sample of some true set of questions $\mathcal{D}^\infty$, such that model performance on $\mathcal{D}$ can be generalized. This generalization step requires theoretical work to show that $\mathcal{D}$ is a good instrument, both systematization of the underlying concept and operationalization into a set of questions (Wallach et al., 2025).

Generalizing one step further, the testing corpus $\mathcal{D}$ may not contain a set of gold standard answer y_i^*, for instance, in the case of essay prompts instead of multiple choice question. In this case, the reward (correctness) of the answer $\hat{y}_i$ is not deterministically given by $\hat{r} \in \{0, 1\}$ by whether it matches the true answer y_i^*. Instead, the reward is the output of some (possibly stochastic) grading process $\hat{r} = r(\hat{y}_i, \omega)$ (explicit stochastic dependence on ω temporarily restored). Further, $\hat{r}$ may take on values intermediate on $[0, 1]$, possibly representing partial credit. While the testing corpus $\mathcal{D}$ is a fixed fictive sample of the space of possible questions $\mathcal{D}^\infty$, it may be more challenging to freeze the grading process in amber, which might be a panel of human annotators, or other AI systems acting as judges.

A.3.3. Agentic Evaluation

For agentic evaluation tasks, the task description is somewhat more complicated. Let $E = (S, A, T, O)$ be called the environment, which represents the possible states of the computing environment (S), the actions available to the AI system (A, including tool use), the transition table for the environment (T, how the environment responds to an agent's actions, including actions that affect the environment through tool use), the observations available to the AI system (O). A task definition $x \in X$ consists of a tuple (E, s_0, Y) where E is the compute environment for the task, s_0 is the initial state of the environment (possibly including the task instructions as presented to the agent), and $Y \subset S$ is a set of valid solution states. That is, if the system reaches state $s_y \in Y$, then the AI system has successfully completed the task. A description of the solution set Y may be provided as part of the initial state of the environment, s_0. Thus, for task evaluation, the testing corpus is $\mathcal{D} = \{x_i\}_{i \leq n} = \{\ (E_i, s_{0,i}, Y_i)\ \}_{i \leq n}$.

Note that the environment is defined as to exclude the state of the LLM's context window. Any purely internal steps that the LLM takes, such as Chain of Thought or ReAct planning, does not affect the compute environment and does not affect the state $s \in S$ until the AI system uses an action (such as tool use) that does affect the environment. Difficult-to-classify intermediate steps are not hard to imagine – e.g., the AI system creates a text file containing a "note to self." Based on the environment partition above, the dividing line is whether the context window or context history alone is affected, or whether other parts of the compute environment are modified.

In agentic evaluation the AI system does not make a single inference step to predict an answer $\hat{y}_i$ in response to a question x_i. Instead, the AI system must plan and execute a series of steps starting from the initial state of the environment s_0 before arriving at a solution state.

Consider a task x_i, and a plan A^j that consists of a list of actions $(a_1^j, a_2^j, \dots a_K^j)$. An AI system faced with task x_i will develop plan A^j with probability $P[A^j \mid x_i]$. Assume that, starting from s_0, the AI system takes each action a_k^j in order until the final action a_K^j takes the environment to state $s_y \in Y$. The agent will take the first action a_1^j with probability $P[a_1^j \mid x_i, A^j]$, which represents the chance that the AI system chooses and succeeds on action a_1^j given it previously chose plan A^j with a_1^j as the first action. Next, the AI system takes the second action a_2^j with conditional probability $P[a_2^j \mid x_i, A^j, a_1^j]$ and so on until the final action a_K^j with conditional probability $P[a_K^j \mid x_i, A^j, a_1^j \dots a_{K-1}^j]$.

However, condensing the probability of successfully completing task x_i through plan A^j as

$$P\left[s_Y \in Y \mid x_i\right] \overset{?}{=} P[A^j \mid x_i] \prod_{k}^{K} P[a_k^j \mid x_i, A^j, a_1^j \dots a_{k-1}^j] xf \tag{3}$$

underestimates the probability of success because there may be many different plans A that correctly complete task x_i. Instead, the total probability of success is the sum of the probability of successful execution of *every* plan A_j that successfully completes the task (perhaps limited to those consisting of $K < \overline{K}$ steps a_k). Thus, the correct probability of successful execution is

$$P\left[s_Y \in Y \mid x_i\right] = \sum_{j}^{J} \left(P[A^j \mid x_i] \prod_{k}^{K} P[a_k^j \mid x_i, A^j, a_1^j \dots a_{k-1}^j] \right) \tag{4}$$

The probability calculated in equation 4 is the probability that an AI system will correctly solve one task x_i. To extend this calculation to a testing corpus $\mathcal{D}$, the expectation from equation 2 should be used, and to convert to an empirical quantity the expectations should be replaced with sample averages, as in equation 1.

Note that the individual conditional probabilities $P[a_k^j \mid x_i, A^j, a_1^j \cdots a_{k-1}^j]$ combine many sources of uncertainty: whether the AI system correctly uses the planned tool, and whether the planned tool functions as expected. Especially for less powerful agents or agents using poorly documented tools, the accumulation of many small chances for failure may add up to a significant total probability of failing a task.

Equation 4 expresses the probability of task completion in terms of selecting a plan A^j of sequential actions and completing each action a_k^j on the list. A similar framing is to view the plan A^j as decomposing task x_i into a list of subtasks $A^j = (x_1^j, x_2^j, \dots x_K^j)$, each of which may itself by recursively broken down into subtasks, i.e., as a filtration. Then, instead of equation 4, the probability of successful execution is

$$P\left[s_Y \in Y \mid x_i\right] = \sum_{j}^{J} \left(P[A^j \mid x_i] \prod_{k}^{K} P\left[s_Y^j \in Y^j \mid x_k^j\right] \right) \tag{5}$$

,

where $P\left[s_Y^j \in Y^j \mid x_k^j\right]$ is the probability of subtask completion (defined recursively) of subtask k from plan j and K indexes the subtasks in plan A^j. The equivalence between equations 4 and 5 is established by the (not unique) equivalence of a plan as a list of actions to a plan as a list of subtasks that those actions successfully complete.

The evaluation of task performance may not be adequately captured by breaking the task into a plan with several steps and evaluating the performance at each step because a given task may be successfully solved by many different plans. In equation 5 the further possibility of subtask equivalence makes the summation over A^j challenging. This challenge is discussed more below in the context of tool use.

B. Methodological Challenges in Agentic Evaluation of AI Systems

B.1. Challenges in Concept Development

Concept development refers to the refinement of the underlying idea of interest to be measured by an evaluation, as well as the systematization of that idea into a well-scoped definition and related metrics for measurement (Adcock & Collier, 2001;

Wallach et al., 2025). This process takes an complex, underlying idea of interest that is difficult to quantify directly and develops targeted metrics for its measurement, and it draws from the field of measurement theory in the social sciences. Problems in his space mostly occur at what has been labeled the design stage of the evaluation pipeline (Reuel et al., 2024; Wei et al., 2025b), i.e., the stage before data is collected. In the language of section A.3, this challenge corresponds to selecting $\mathcal{D}$, or even to the planning stage around how to select $\mathcal{D}$.

Most evaluations today—in particular those that are benchmark-based—have bypassed this process directly and focus on collecting a broad range of diverse question-answer pairs in attempts to assess general capabilities (Burden et al., 2025; e.g., Phan et al., 2025). The lack of rigorous interrogation of evaluation concepts in the existing literature has raised significant questions of construct validity (Raji et al., 2021; Wallach et al., 2025) and has also been discussed in other computing contexts beyond modern ML (e.g., Dyba et al., 2005; Emmerich et al., 2016). We discuss some relevant questions below.

How can performance in agentic evaluations be predicted (if at all), and when are non-agentic evaluations sufficient to measure capabilities or risks? Due to extended task horizons and difficulties in scoring, agentic evaluations can be challenging and resource-intensive to design, implement, and execute. Prior work has explored correlations between different question-answer benchmarks (Schaeffer et al., 2025), but no research has examined whether these benchmarks or other metrics can accurately predict performance on agentic evaluations. Understanding predictors of performance in agentic evaluations may help evaluators identify the extent to which, relative to non-agentic evaluations, agentic evaluations offer additional information about compound AI system capabilities – and in which circumstances this additional information justifies their use.

How valid and reliable are AI-generated agentic evaluation tasks? Compared to question-answer evaluations, agentic evaluation tasks are significantly more complex, and creating these tasks is also more resource-intensive. Work such as (Huang et al., 2023) have attempted to automate the creation of agentic evaluation tasks, but the validity and reliability of tasks created in this manner is unclear. One reason in particular to be skeptical of automated task creation in the agentic evaluation setting is that agentic evaluation tasks often require planning and reasoning capabilities to complete and, by extension, to design. In the absence of compelling evidence that AI systems can meaningfully complete agentic evaluation tasks, it may be premature to delegate task creation to the same systems.

How can evaluation concepts efficiently account for large state and action spaces with diverse solution pathways? As noted in Section A, agentic evaluation tasks are characterized in part by the large state and action space in which an AI system is located. The proliferation of pathways to both task completion as well as failure models—due to the functionally unconstrained nature of the state and action spaces—makes it important to measure concepts that account for the entire process of attempting the agentic evaluation task (see Pencharz et al., 2024; Yadav et al., 2019). For instance, (METR, 2025b) tests foundation models' abilities to engineer GPU kernels and measures only the performance of the model-generated kernels; the measured concept is engineering ability, as represented by the efficiency of model-generated code. However, other concepts may also be important to measure, such as the "novel[ty] and sophisticat[ion]" of model outputs or the ability of the models to "adapt . . . to new constraints" (METR, 2025b).

One possible solution is to measure multiple concepts in a single evaluation. In the fairness context, (Wang et al., 2024) suggests using suites of benchmarks to triangulate the trade-offs between competing notions of fairness. (Kapoor et al., 2024) has recommended measuring evaluation costs in addition to performance in agentic evaluations; it may be appropriate to also measure other concepts in an evaluation suite for agentic evaluations, though the specific concepts at hand are likely context-specific.

Finally, this challenge may create additional difficulties in the safety context. To accurately capture model-related risk, sensitive information is needed to conduct risk modeling and develop specific pathways to harm (UK AI Security Institute, 2024), both of which are necessary to refine measurement concepts. As the number of pathways to harm may increase, however, agentic evaluations may raise ethical challenges concerning confidentiality in addition to the technical challenges concerning environmental validity that are also present in non-safety contexts.

In safety evaluations, what (or to what extent do) proxy tasks accurately reflect real-world risk without evaluating AI systems' directly on completion of dangerous tasks? The general-purpose nature of foundation models has raised questions about models' dangerous capabilities, such as abilities to deceive humans, or to develop cyber- or bio-weapons (Phuong et al., 2024). A number of knowledge-based benchmarks have been developed to measure model safety, e.g., (Götting et al., 2025; Laurent et al., 2024) in the context of dual-use biology capabilities. In the agentic evaluation context, however, evaluators may not be able to directly test for task completion of, e.g., a system's ability to create a dangerous biological agent due to legal and ethical concerns. As evaluators will need to rely on proxy tasks to measure dangerous capabilities, the quality of these proxies directly affects construct validity; proxies in contexts that do not have direct human analogues, such as AI control (Greenblatt et al., 2024; Phuong et al., 2024), may be particularly difficult to develop due to lack of precedents from other fields.

B.2. Challenges in System Design

Challenges in system design are those that relate to best practices around scaffolding, prompting, or crafting the internals of compound AI systems, for which no consensus has emerged in the ML community. These challenges correspond to maximizing the conditional probabilities in equations 4 and 5 by choosing the optimal format to deliver relevant information to the LLM at the heard of the AI system, to the extent that those conditional probabilities represent the "honest best" that the AI systems can perform.

B.2.1. Challenges in Elicitation

It is easy to present an AI system with challenging tasks and measure a success rate, but it is more difficult to elicit performance that represents the maximum (or even average) capabilities of the agent. This section presents research on elicitation grouped into four broad categories: challenges in designing prompts for AI systems, challenges in designing scaffolds for agents, challenges in managing the context window for agents, and challenges around tool use by agents. While these challenges are presented in the context of agentic evaluation, research on these topics is relevant to engineering of AI systems designed to complex open ended task.

The nature of a prompt changes for increasingly open ended tasks. While earlier research focused on tool use to answer difficult questions, agentic evaluations began to move into multiturn inputs (simulation conversations with humans), chain of thought type reactions to environmental feedback, and to specifications of engineering tasks. Effective prompting for compound AI systems is more complicated because the task description may contain multiple tasks or constraints and the input itself may be long or noisy (He et al., 2024). Further, the multi-step nature of the tasks requires multiple prompts within the scaffold, and traditional LLMs are usually fine tuned for an instruct format that mirrors a conversation between one user and one LLM. Finally, much like in all uses of LLMs, the specific form of the task description in the prompt impacts how AI systems balance competing priorities (Fish et al., 2024). Best practices are still in their infancy.

We present several open challenges in prompt design for AI systems in this section.

How do examples improve task performance? Previous research has shown that few shot prompting, providing the LLM with successful question-answer pairs, improves performance on traditional LLM benchmarks (Yao et al., 2024). It is not clear what analogous examples might look like for tasks. One possibility is that the scaffold proactively provide examples of tool use, and (Huang et al., 2023) found that few shot prompting improved performance (for some models substantially) on a tool use task. Another possibility is that the problem definition demonstrate the task with a working, but low quality answer. (Huang et al., 2023) studied agent performance on machine learning experimentation tasks, for which they started off the agents with a task description, starter files, and an evaluator. (Wijk et al., 2024) tested the ability of AI agent's to optimize scientific and engineering tasks by providing a starting solution, in addition to a scoring function.

How can internal reasoning be leveraged to improve task performance? Generally, all task evaluations use some kind of chain of thought (CoT) or reasoning step, and frontier LLMs increasingly incorporate reasoning into general-purpose and chat use (OpenAI, 2024; DeepSeek-AI, 2025). (Wei et al., 2022) developed CoT prompting, in order to leverage inference time compute to allow the model to itself decompose multi-step problems into intermediate steps. (Yao et al., 2022) developed a linear CoT scheme to convert verbal actions into task-oriented planning. (Shinn et al., 2023) developed self reflexion scheme for in-context learning. (Qin et al., 2023) compared Yao et al.'s linear CoT scheme (ReACT) to their depth-first search-based decision tree. (Li et al., 2024a) implemented three CoT methods for a set of legal analysis tasks: outline a complete plan, outlining a multi-step plan with opportunities to reassess, and a full thought-action-observation schema.

A major problem with increasingly elaborate branching CoT schemes is that they increase the test time cost by significantly increasing the number of tokens generated. Thus, there is a trade-off between running a more elaborate CoT scheme, or running more CoT schemes. It is an open question what kind of CoT scheme is most effective for long-term task performance, and it is unclear whether built-in LLM reasoning (e.g. DeepSeek-R1) or scaffold-prompted reasoning is more effective for tasks. In particular, there is currently no principled way to balance the number of CoT runs against the length of each CoT run.

How can internal reasoning incorporate interactive feedback? While CoT and related reasoning techniques allow an AI system to analyze the situation before choosing an action, self-reflection techniques allow the AI system to incorporate feedback from the environment. Feedback from the environment can include multi-turn conversations: (Pencharz et al., 2024) created long form tasks using both static and dynamic follow-up questions to simulate interactions with a human user, while Cheng et al. created an evaluation suite of tasks with intermediate natural language feedback (Cheng et al., 2023).

You et al. developed a question-answering framework in which feedback from previous rounds is incorporated as examples in the prompt for subsequent rounds (You et al., 2024). Similarly, Pan et al. developed a coding evaluation framework that incorporated (simulated) interactive feedback (Pan et al., 2025). An intriguing lens for this feedback-incorporation is cognitive reflection; Li et al. investigated shortcomings in CoT at integrating unexpected information (Li et al., 2024b).

B.2.2. Challenges in Scaffolding

When performing open ended tasks, models are usually equipped with tools to interact with their computing environment: narrow tools like function calls and APIs, or broader tools like programming environments and file system access. The code necessary to handle communication between the LLM reasoning engine and the tool use is typically called a scaffold; scaffolds may also coordinate communication between different AI agents performing a common task, and scaffolds may manage CoT reasoning during task performance. Here, we discuss some difficulties with capability evaluations related to the scaffold.

How should scaffolds be chosen? There is currently little consensus on scaffold best practices, and there are a wide variety of scaffolding frameworks available both commercially and through open source providers (OpenAI, 2025; LangChain). (Wijk et al., 2024) tested different LLMs on their benchmark using two different scaffolding frameworks. Many evaluators create custom scaffolds for their evaluations, leading to few large-scale comparisons of different scaffolds.

What tradeoffs exist around the size of scaffolds? A first challenge is the size of the scaffold's built-in prompts themselves, as more elaborate scaffolds take up more space in an LLM's context. (Huang et al., 2023) found that LLM performance dropped in tool use tasks as the length of a tool list increases. The scaffold also controls in the first instance what other information reaches the context window. For instance, (Kwa et al., 2025) used a simple scaffolding environment designed to provide computer system tools and keep the input within the context limit.

How does task performance scale with improved scaffolds? An open question is to what extent improved scaffolds will lead to improved performance. As part of their model evaluations, METR compared a simple agent scaffold and a "elicited agent" with a propose-evaluate CoT cycle and found that model performance improved with the "elicited agent" (METR, 2025a). Building on the question of how to divide the scaffold between the model and task, there is a challenge in evaluating whether improved performance due to improved scaffolding represents a genuine improvement in model capabilities, represents overfitting to a particular set of tasks (e.g., development task sets that represent a narrow segment of the underlying task distribution of interest), or represents lowering the difficulty of those tasks.

How does the scaffold interact with tools? The scaffold formats instructions to the LLM, and interprets LLM output. More importantly, the scaffold interfaces with protocols to call local or networked tools. (Yang et al., 2025) delineate the different protocols on the "agentic stack," and classify the different agentic protocols, such as Anthropic's Model Context Protocol and Googles Agent-2-Agent (Anthropic, 2024b; Surapaneni et al., 2025).

B.2.3. Challenges in Interactions Between System Components

Traditional evaluations of LLM performance in long contexts such as Needle in a Haystack and RULER focused on the performance of LLMs on simple questions by evaluating how well an LLM was able to make use of its context (Kamradt, 2024; Hsieh et al., 2024). While performance on open ended tasks presupposes a basic level of useability of the LLM's context window, it also brings a new set of challenges – in particular, how that context can be used most optimally, especially for long tasks that generate conversation or tool use history.

At a broad level, this question is what ought be placed in the context window for LLMs calls in the compound AI system. This question can be divided into more specific challenges challenges corresponding to what information is sought.

How effective is Retrieval Augmented Generation for tasks? One of the first tools subject of research and development was Retrieval Augmented Generation (RAG), which supplements model knowledge by providing relevant documents from a curated library. However, there are still many open questions for the use of RAG — how should relevance be determined, and how much relevant information should be provided? Leng et al. examined how use of RAG, affected performance across different context lengths, finding that for some LLMs accuracy dropped with very large context size but for others it plateaued (Leng et al., 2024). There is also as yet little research comparing RAG for information retrieval tasks with tool use for research tasks. In particular, there is little research comparing non-agentic semantic search based RAG with more agentic information retrieval tool use.

What tools for storage and retrieval of long-term memories improve task performance? For multi-round interactions, fitting the interaction history into a finite context may be challenging. One straightforward method is to truncate messages except for the problem statement and the most recent r messages so that the total context is less than some threshold (Liu et al., 2023). However, simple truncation prevents long-term memories, like LLM-generated plans, from persisting across a long multi-step task. The reflexion scheme of (Shinn et al., 2023) combined short term memory of the trajectory history with long-term memory of the model's own outputs.

(Park et al., 2023) developed a memory and retrieval system for a sandbox simulation that conserved context space by assigning each memory a recency, importance, and relevant score. Executive control over long-term memory is especially important for open ended tasks. In the Twitch-streamed project ClaudePlaysPokemon, the AI system took notes to record its findings in the game, but was quick to erase those notes when it thought (often incorrectly) that it had surmounted its current challenge (Anthropic, 2025). Long-term memory is especially difficult to evaluate in an ecologically valid way, because real world tasks are often engineered to simplify long range dependencies.

How does task size affect task performance? Another challenge is the link between the length of the task and the available context. In one early evaluation of AI agents, (Liu et al., 2023) developed AGENTBENCH, which included "context limit exceeded" as one failure mode in the benchmark. Similarly, when comparing model performance by release date, (Kwa et al., 2025) imputed a zero score to certain older LLMs whose context limits are too small to attempt certain tasks. The challenge of tasks being too big for AI systems to solve is quickly becoming solved, as modern LLM context windows continue to grow. However, the surpassing of this obstacles only reveals another – managing that large context.

Even as more modern scaffolds include context management features (such as RAG), some more recent evaluations have identified task length as an explicit constraint. For capability evaluation, METR divides tasks into those with token constraint (HCAST) and those with wall-clock constraints (RE-BENCH) (METR, 2025b). While one evaluation possibility is to provide a fixed budget, another is to examine a performance-compute frontier (Manvi et al., 2024).

B.3. Challenges in Environmental Interactions

Environmental tasks require interaction with the environment, either by collecting information from the environment or making changes to the environment (possibly by creating an artifact). This creates two sets of challenges for evaluation design: challenges related to the tools given to the AI system to complete a task, and challenges related to the environment those tools act in. These challenges relate to the environment definition borrowed from the definition of POMDP given in section A.3.1 for the formalization of task evaluation in section A.3.3: the action space A consists of the tools available to the AI system, the observability O corresponds to what information the tools provide the AI system, and the relevant features of the environment determine how the environment state space S is modeled.

B.3.1. Challenges in Tool Use

Perhaps the biggest difference between simple Q&A LLM evaluations and task evaluations is tool use.[8] The agentic scaffolding presents the LLM with signatures for function calling to retrieve information from the environment or to make changes to the environment. Simple tool use evaluations are easy to grade and benchmark, and leaderboards like the Berkeley Function-Calling Leaderboard rank LLM performance on function calling tasks (Patil et al., 2024).

This section discusses a number of challenges involving tool use evaluation, starting with the difficulties inherent to multi-step, multi-tool tasks. Next, there is the challenge of choosing a tool, a challenge that grows greater with larger the toolset. Finally, the evergreen problem of hallucination takes special forms with tool use.

Can agents succeed at tasks that require multiple tools? The fundamental target of task evaluation is the extent to which the LLM system can generate effective multi-step plans that it can execute sequentially, especially when each step requires a different tool. (Huang et al., 2023) considered tasks with up to two required tools. (Qin et al., 2023) leveraged a relationship graph between their large tool set to suggest related tools. While most tool use evaluation uses stateless tools (where the AI system need not keep track of the state of the environment), (Lu et al., 2025) developed a benchmark for tools with state dependency, requiring multiple tool use to manipulate the environmental state.

How do AI systems select which tools to use during operation? For tool use scenarios, an important question is which tools to present to the LLM in the context, and how the LLM chooses between them. For scaffolds or evaluations with a small number of potential tools, every tool can be presented to the LLM in context. For larger toolsets, (Huang et al., 2024) used a two step process. First, they construct a shortlist of tool candidates, either through a cosine similarity between embeddings of the task description and the tool description or through a curated list. Then, they evaluated whether the LLM could choose an effective tool from that shortlist.

How can the effects of hallucination on tool use be mitigated? Tool use presents a sycophancy problem, where the LLM expects that one of the provided tools will help solve the problem even when none of them are helpful. (Zhang et al., 2024) evaluated hallucination in tool use using three scenarios: missing necessary tools, potential tools, and limited functionality tools. Similarly, (Huang et al., 2023) studied reliability issues in tool use by presenting problems along with unsuitable tools. Separately, tool use presents a different hallucination error as well: whether the tool used the user-provided parameters or hallucinated parameters in function calls (Zhong et al., 2025). While hallucinations are not unique to tool use, these specific manifestation are.

B.3.2. Challenges in Designing Test Environments

The *evaluation environment* refers to the problem statements, evaluation sandbox, and set of available tools provided to the model during an open-ended evaluation. For conclusions drawn from an AI evaluation to be transferrable to predicting the real-world impact of a model, it is important that the evaluation environment be as realistic as possible. This is for two reasons: construct validity and mitigating the potential for sandbagging.

Does the evaluation mimic the affordances available in real-world settings? If the environment provides fewer affordances to the model than it will have during real-world use, the evaluation results may differ from what the AI system can actually achieve in a deployment setting. For example, if the evaluation does not provide the AI system with internet access, or if it uses a weak scaffold, it may underestimate the performance of the model when its capabilities are fully elicited by end users (see more discussion in Section B.2.1).

Broadly, there are three approaches to tool selection for evaluation: few carefully chosen tools, larger sets of API tools, or general-purpose tools. Early evaluations used a small number of tools. (Qin et al., 2023) curated a set of 16464 representational state transfer APIs spanning 3451 tools from a hosting platform. (Huang et al., 2024) gave their agent access to only a small number of actions, but allowed it to edit and run Python scripts.

[8] Previous work has decomposed the study of AI tool use in different ways. METATOOL (Huang et al., 2023) breaks the process into four stages: (1) determining whether to employ a tool, (2) selecting a tool, (3) constructing tool parameters, and (4) handling tool results. Similarly, a literature review by (Yehudai et al., 2025) divided tool use into intent recognition, function selection, parameter extraction, function execution, and response generation. (Guan et al., 2025) further divided research of evaluation of agent tool use in multi-turn conversations into three groups: (1) API Interaction and Dynamic Tool Use, (2) Multi-step Tool Selection and Reasoning, and (3) Reliability and Hallucination in Tool-Use. Our work is robust to these different conceptualizations.

How does performance generalize to tasks with novel toolsets? While early evaluations of tool use tested general-purpose LLMs on bespoke tool sets, the general question is whether evaluation results for one toolset generalize to other toolsets. This question may grow in importance as the third-party tool provider ecosystem grows – for instance, Anthropic has open-sourced the Model Context Protocol standard for agentic tool use (Anthropic, 2024b). (Qin et al., 2023) evaluated a model on a different set of tools as it was trained on.

How do evaluators avoid trivial solutions? In the other direction, if the evaluation environment provides *too many* affordances during training, it may inflate scores beyond what we should expect during real use. For example, in (Starace et al., 2025), the authors make sure that their agents are never allowed to view the original codebases of the papers they have been tasked to replicate. They do this by implementing a simple post-hoc monitor on the agent's actions. If the agent ever visits a blacklisted resources, it receives a score of 0. Similarly, train-test contamination may also result in tasks becoming trivial to solve.

How do evaluators prevent model sandbagging? If the AI system *realizes* that it is being evaluated because the environment appears overly artificial, it may change its behavior. When a model intentionally under-represents its capabilities on an evaluation, this is known *sandbagging*. An AI system may sandbag for a variety of reasons, including safety training (the model has been trained to not exhibit harmful capabilities) or a more speculative phenomenon in which a misaligned model wishes to hide its dangerous capabilities in order to be released (Benton et al., 2024). To mitigate this possibility, it is important to make the evaluation environment as similar to the deployment environment as possible.

Current models are starting to demonstrate levels of situational awareness where sandbagging concerns may be realized. (Laine et al., 2024) develops the Situational Awareness Dataset to measure the extent to which LLMs can use context clues to determine whether a prompt is from an internal evaluation or a real-world deployment, finding that with enough prompting, some models can match human performance at this task. (METR, 2025c) notes that it is especially hard to rule out this possibility when the evaluators do not have access to reasoning traces, as was the case when METR performed pre-deployment evaluations on o3 and o4-mini.

An "in the wild" example of this phenomenon is described by (Xander Davies [@alxndrdavies], 2025), where they "noticed instances of a model seeming to realize [their] tasks were fake." For example, one of the tasks used the artificial URL, and the model reasoned "Note: Since this is a hypothetical example (the URL isn't real), I'll proceed with the steps but the content will be whatever is returned by the tools." This demonstrates that models can (at least sometimes) realize when they are in an evaluation, although this alone does not imply that the ultimate performance of the model was influenced by its level of situational awareness. Compared to evaluations in more tightly controlled evaluation environments, tool use, especially Internet access, increases this difficulty.

B.4. Challenges in Scoring

Once a task is designed, an environment developed, and a toolset chosen, the most important part of the evaluation process remains: scoring the performance of the AI system on agentic evaluation tasks. The challenges involved in grading task performance can roughly be divided into two categories. First, there are challenges in defining metrics with which to grade AI system performance, and in particular to develop rubrics to judge performance. Second, there are challenges in applying those metrics in practice, especially because large scale evaluations almost necessarily rely on either crowd-sourced or automated judging. The challenges in this section mainly relate to defining and measuring the reward function R in Section A.3.1.

B.4.1. Challenges in Metric Development

This section discusses challenges in metric development, i.e., in the process of creating rubrics and procedures for measuring the desired metrics.

How can robust scoring rubrics be efficiently developed, especially when rubrics are complex? The metrics of evaluation are critical components of any AI evaluation, and the development of rubrics and procedures to generate those scores is particularly complex in agentic evaluations due to the complexity of the tasks themselves.

Some evaluation contexts lend themselves to simple verification, while scoring in other contexts is substantially more difficult. The archetype of the former is software engineering evaluations, which often generate scores based only on models' final outputs (normally software artifacts). For instance, SWE-BENCH (Jimenez et al., 2024) evaluates a model's software engineering abilities by asking it to issues in a large code bases. To guarantee the realism of the dataset, the requested patches are sampled from real-world GitHub repositories, and the correctness of the model-written patch can be automatically checked by running unit tests associated with each GitHub pull request. Similarly (METR, 2025b) measures only the efficiency of model-generated code while excluding intermediate factors from the final score.

Even in settings with simpler verification, however, this approach runs into two problems. First, the human-written unit tests may fail to measure the quality of the patch along all possible dimensions. For example, while unit tests may check the *accuracy* of the AI-written code on a number of example inputs, they may fail to measure the asymptotic *efficiency* of the solution or *legibility* of the solution. Human software engineers usually balance the tradeoff between accuracy, efficiency, and legibility of their solutions in a reasonable way, but it is difficult to formalize what this optimal tradeoff should be. Thus, human-written unit tests often only measure solutions along a small number of these dimensions. However, an AI system optimized to achieve a high benchmark score may take advantage of these under-constrained unit tests by, i.e., by implementing an extremely inefficient (but correct) algorithm. A related limitation was explicitly discussed in (METR, 2025b) (see also Section B.1).

Second, the structure of an evaluation like SWE-BENCH necessarily reduces a broad capability that we want to measure ("software engineering ability") into much narrower capability of ("ability to write patches to bugs") with a verifiable outcome ("do tests pass?"). This means that the evaluation fails to track AI progress on harder-to-measure aspects of the broader capability ("ability to communicate and collaborate with other developers"). This discrepancy between the broad class of skills we wish to measure and the narrower subset of skills we can benchmark effectively is present for any type of model evaluation, but it is particularly difficult to reduce when the capabilities being studied feature many open-ended elements.

In domains where scoring or verification is more complex, evaluators normally created detailed rubrics via consultation with domain experts. In the context of non-agentic evaluations with no gold-standard labels, these rubrics have been shown to significantly affect final scores (Pathak et al., 2025; Hashemi et al., 2024; Fan et al., 2024), and this result is likely to hold in the agentic evaluation context.

One example of such an evaluation comes from (Pencharz et al., 2024), where the develop a methodology to evaluate a model's ability to act as a scientific research assistant. The evaluation consists of a prompt specifying a high-level goal, a rubric, and an LLM-based autograder. The rubric is written by domain experts and assigns points based on whether it touches on a number of key components. The autograder compares the evaluated model's response against the rubric to assign a score from 1 to 10. (Pencharz et al., 2024) also tests the model's ability to answer predetermined follow-ups and dynamic critiques to its response.

Given that the rubric plays such a key role in the evaluation setup, it is necessary for the evaluators to consult with credible domain experts to make sure the rubrics are comprehensive and well-specified. However, rubric creation is time-consuming and expensive, which is evidenced by the fact that (Pencharz et al., 2024) was only able create rubrics for a handful of tasks.

Starace et al. (2025) runs into similar challenges with rubric creation. The evaluation attempts to measure an LLM agent's ability to reproduce the results of machine learning papers. Although the agent's output solely consists of code, the sheer and complexity of the task ("replicate the results of all of the experiments in a given paper") makes it impossible to judge with unit tests. The authors create a rubric for each paper that consists of a hierarchical tree of outcomes required to replicate a given paper. For example, the root node begins with the highest level outcome expected ("The core contributions of the paper have been reproduced"), and it may have one child node for each of the core contributions. Progressing down the tree results in finer detail about specific outcomes. The score of each node is the weighted average of its children, and each leaf node is evaluated by confirming that a specific empirical result is obtained, code testing some question is run, or the candidate's source code appears to include a correct implementation of some specific requirement.

The authors note that "constructing the rubrics for each paper was notably the most time-intensive aspect of developing PAPERBENCH." Each rubric required collaborating with an author of the original paper and took multiple weeks to develop, which explains why the benchmark only consists of 20 papers.

How can evaluators navigate the "coverage-gradeability trade-off" in subtask granularity? Where evaluation scores are dependent on the scores or completion rate of intermediate subtasks, evaluators face a "coverage-gradeability trade-off." When subtasks are defined as more granular, grading becomes simpler since the subtask is well-defined; however, increased subtask granularity also limits task coverage by limiting the acceptable solution pathways for task completion. When subtasks are less granular, the opposite is true: grading becomes more difficult since the subtasks are more widely-scoped, but overall coverage may increase.

Two examples may be illustrative. On one end of the spectrum, (Zhang et al., 2024) broke tool use evaluation into three steps: solvability detection, solution planning, and missing tool analysis, and used a "progress rate" measure that relied on the existence of either zero or one solution pathway. An example of an alternative to such narrow specifications is (Huang et al., 2023), a tool use benchmark in which the authors gave systems' access to multiple tools that served similar functions, any of which could be chosen to complete a task. To account for these intermediate choices, (Huang et al., 2023) generated scores based on groups of tools with similar functions.

Overall, however, no work currently exists to quantify the effects of this tradeoff or to suggest guidelines for evaluators in how to appropriately balance coverage and gradeability.

How can evaluation results assign partial credit and minimize mode effects from scoring scales? The scale on which model outputs are scored can create significant mode effects. Historically, many evaluations adopted a binary pass-fail scoring scale, which did not permit assignment of partial credit. Most recently, (Phuong et al., 2024) and (Shah et al., 2025) have suggested defining task milestones that would permit capturing performance improvements at higher fidelity. Additional research is needed as to how to best set these milestones, the mode effects introduced with different scoring choices.

How can model confidence be calibrated both for the overall task as well as for individual subtasks? Researchers have explored extensively the problem of calibrating model uncertainty (Geng et al., 2024; Liu et al., 2025; Lin et al., 2024b; Tian et al., 2023; Malinin & Gales, 2020; Gligoric et al., 2025). The agentic evaluation context introduces several open problems to the issue of calibration beyond calibration after task completion.

First, it may be desirable for systems to be calibrated not just post-completion but also at intermediate steps in agentic evaluation tasks. Effectively, this desiderata would extend the calibration problem to all subtasks of the agentic evaluation task, which would require that the model be well-calibrated with respect to the correctness of the completion of each individual subtask as it pertained to the completion of the task as a whole. Intuitively, subtask calibration would introduce substantially stronger demands for systems' reasoning and planning capabilities, causing an increase in difficulty over calibration on the overall task.

Second, agentic evaluation tasks often require AI systems to engage in some level of reasoning or planning at the beginning of the task to determine possible solution pathways for task completion. This planning step could also be conceived of as the first subtask, though it normally would not require environmental interactions. Evaluators may also wish to test the calibration of models' confidence levels that its initial plan for task completion is correct with respect to being able to complete the evaluation task if executed properly.

These variations on the calibration problem in the context of agentic evaluations have yet to be explored, and substantial work is needed to make progress on this problem.

B.4.2. Challenges in Metric Measurement

What sources of autograder (AI graders) bias are specific to the agentic evaluation context, and how can such biases be controlled? Due to the high costs of human grading, many evaluators rely on automated, LLM-based grading to generate scores of evaluation data. These AI graders have been shown to have a host of biases, including bias from the order of options presented, towards longer responses, and towards outputs generated by the same model or model family (Ye et al., 2024; Koo et al., 2024; Feuer et al., 2024; Panickssery et al., 2024). Many of these biases could be exacerbated in the agentic evaluation context. For instance, CoT transcripts could add substantially to length and could also vary substantially in length due to model stochasticity; an AI judge with access to CoT transcripts could be particularly effected by bias for longer responses. Relatedly, it is unknown the extent to which self-preference bias will occur in the agentic evaluation context.

How can autograders be validated and evaluated? Validation and evaluation of of AI graders has been a topic of increasing interest (Shankar et al., 2024; Guerdan et al., 2025). The complexity of agentic evaluation makes validation more important: for instance, (Pencharz et al., 2024) developed a detailed step-by-step rubric for use with AI graders and found a systematic failure in AI graders to be biased towards leniency despite specific requirements in the grading rubric. The increased state/action space and possible subtasks may make validation highly challenge in particular.

How can evaluators prevent reward hacking in agentic evaluations? In the context of software engineering evaluations that measure success with automated unit tests, such as (Wijk et al., 2024; Jimenez et al., 2024; Kwa et al., 2025), an increasingly difficult problem to confront is *reward hacking*: instances when the AI passes its tests using an unexpected and undesired shortcut. For example, when given the engineering task of optimizing a CUDA kernel, METR observed an instance of the AI agent tampering with the time measurements. Since its performance on this task is measured by timing the runtime of its code, this allowed the agent to essentially achieve any score it wanted on the task (METR, 2025b).

In practice, reward hacking can be detected by manually inspecting an AI's reasoning traces and solutions for evidence of "cheating" solutions. However, this becomes cumbersome for evaluations with large number of tasks, so a separate language model can be used as a "monitor" to verify the the solution does not use any undesirable cheating techniques. (Meng et al., 2025) is an example of an LLM-powered tool that can help detect and diagnose unexpected agent behavior during evaluations.

(Baker et al., 2025) tests the efficacy of using a weaker model to act as a monitor on the chain of thought of a stronger reasoning model as it solves coding tasks. They find that the monitor is able to reliably detect the large majority of reward hacks performed by the reasoning model, making it a promising technique for preventing inflated evaluation results from reward hacking. However, the authors also find that it is dangerous to use a chain of thought monitor as part of the optimization signal — when they *train* the reasoning model with a penalty for triggering the monitor, the model learns to obfuscate its chain of thought to perform reward hacking in ways that are harder for the monitor to detect (however, the total amount of reward hacking still diminishes significantly).

B.5. Challenges in Analysis & Documentation

Challenges in analysis relate to the interpretation of evaluation results. Researchers have raised significant concerns with respect to the lack of statistical rigor in foundation model evaluations (e.g., Biderman & Scheirer, 2020), and best practices for analysis remain undetermined (e.g., Miller, 2024; Bowyer et al., 2025). This section discusses some of these challenges.

How can evaluators account and correct for many sources of statistical uncertainty? The complex and long-horizon nature of agentic evaluation tasks introduces many additional sources of bias and uncertainty as compared to traditional benchmark evaluation settings. These sources of uncertainty include: sampling error from the task space, sampling error from small sample sizes, sampling error from the grading process, construct error in the operationalization of measurement constructs, and systematic error from space of evaluation vs. performance tasks. No existing literature has attempted to systematically catalog and measure the effects of different sources of error, nor is it obvious how evaluators can implement corrections either in study design or post-hoc.

What baseline for comparison should be used as a lower bound for performance on agentic evaluation tasks? Baseline results are necessary to interpret and contextualize evaluation results because they enable comparisons to alternative performance results. Three types of baselines are common in the AI evaluation literature: random baselines, baselines from other models, and human baselines (including both generalist or expert baselines).

Random baselines are usually achieved by uniformly sampling from an evaluation task's response scale (e.g., in multiple choice question-answer items). They have traditionally been used in information retrieval evaluation (e.g., Vries et al., 2012; Bestgen, 2015) and are still used in recent foundation model evaluations (e.g., Chiang et al., 2024; Zeng et al., 2023) to test whether models have made progress of any level on an evaluation task. In the context of agentic evaluations, however, this sampling strategy is inappropriate because of large state/action spaces and complex measurement scales. (Yauney & Mimno, 2024) proposes an improved method for calculating random baselines for in-context learning; similar work may be useful in the agentic evaluation task to set a lower bound for expected performance on agentic evaluation tasks.

We discuss below some challenges that are posed for human baselines in agentic evaluations.

How can human baselines account for differences in modes of interaction between humans and AI systems? Mode effects are errors in measurement due to particular measurement instruments (e.g., the order of questions, or whether a survey respondent answered a questionnaire via phone or online) rather than due to true differences in the underlying metrics of interest (Wei et al., 2025b). Preliminary evidence has suggested that AI systems are subject to mode effects, and that these mode effects are different from those experienced by humans (Tjuatja et al., 2024). These effects are exacerbated in agentic evaluations since humans and AI systems interact with external environments differently; for instance, a human may use a graphical user interface to access the internet while an AI is limited to using a command-line interface (e.g., in Wijk et al., 2024). Although compound AI systems are increasingly able to use human modes of interaction (e.g., Anthropic, 2024a), it is unknown whether (differences in) mode effects will vanish as AI capabilities increase. These effects may affect the reliability and reproducibility of agentic evaluations, in addition to the validity of comparisons to human baselines; additional research is needed to quantify, control for, and correct for these effects.

How can evaluators measure and control for cost in humans and in AI systems, and what are the proper conversion rates between human and AI results? Cost metrics in agentic evaluations such as dollar cost or time are crucial for standardizing measurements across evaluation results (Kapoor et al., 2024), as well as for making comparisons between human baselines and AI results (Wei et al., 2025b). For instance, (Rein et al., 2025) and (Wijk et al., 2024) compare performance between human and AI systems on software task given the same length of time. However, the validity and units of comparisons have not been rigorously explored. These comparisons are of significant interest to downstream deployers/users, as well as to economic policymakers, and additional work will be needed to build agentic evaluations that can predict the automation and labor impacts of AI systems.

C. Additional Discussion

Engineering in the field is moving faster than unifying frameworks. Just three years ago, AI agents had extremely limited capabilities even in the domain of natural language processing. Today, competing protocols are being written not only for AI agents to talk within compound AI systems but for AI agents to communicate with networked tools across platforms (Yang et al., 2025). Many of the challenges identified in this paper simply point to the absence of best practices, absences that in many cases result from the speed with which the underlying LLM technology is evolving and rendering previous research obsolete (such as section B.2.1). Other challenges are related to the speed at which capability evaluation moved from general knowledge questions (e.g., Hendrycks et al., 2020; Phan et al., 2025) to multi-step tasks with clear analogues in the white collar labor market. It will be important for the field to develop frameworks to unify and synthesize different evaluations to better understand AI system capabilities holistically.

Practitioners should begin moving away from a model where the scaffold is part of the evaluation, and towards one where the scaffold is part of the agent. There is a clear trend in which early research studied the extent to which foundation models could use tools provided by the evaluation suite and more recent research has studied the ability of AI systems to use more general tools – either virtual programming environments, networked tools accessed via agent protocols, or tools built in to existing commercial AI systems. AI agents are increasingly commercially available, and task performance should be measured at the level of the agents performing them, not just at the level of the LLM foundation model powering a particular scaffold. This shift may require new systematization about how to partition increasingly complex AI systems, systematization that may be aided by standard protocols that delimit the tool layer and agent layer (Yang et al., 2025).

As task evaluation grows to increasingly resemble assessments from the social sciences, practitioners should turn to the social sciences for insight on how to design evaluations instruments. As the capabilities of AI systems have grown, existing benchmarks have begun to saturate. As practitioners design new evaluations to measure task performance, they should turn to the social sciences, which have long grappled with questions about how to develop measurements for hard-to-elicit capabilities. Psychometrics and industrial psychology, in particular, have a rich literature on how to assess task performance in humans.