BRIDGE: Bridging Reasoning In Difficulty Gap between Entities

Fengyuan Liu[1,2 †] **Jay Gala**[1,2 †] **Nilaksh**[2,3 †] **Siva Reddy**[1,2 ‡] **Hugo Larochelle**[2 ‡]

[†] Student [‡] Mentor

[1]McGill University [2]Mila - Quebec AI Institute [3]Polytechnique Montreal

{fengyuan.liu, jay.gala, nilaksh.nilaksh, siva.reddy, hugo.larochelle}@mila.quebec

Abstract

Evaluating the real-world capabilities of AI systems requires grounding benchmark performance in human interpretable measures of task difficulty. Existing approaches that rely on direct human completion time annotations are costly, noisy, and difficult to scale across benchmarks. In this work, we propose BRIDGE, a unified framework that aligns task difficulty for humans and models by anchoring latent difficulty estimates from Item Response Theory (IRT) to human completion time. Using a two-parameter logistic IRT model, we jointly estimate task difficulty and model capability from binary performance data across multiple benchmarks, including METR, SWE-bench Verified, MLE-bench, and GDPval. We show that latent task difficulty varies linearly with the logarithm of human completion time, enabling reliable estimation of human duration for tasks without explicit annotations. Leveraging this alignment, we forecast frontier model capabilities in terms of human task-length horizons and find exponential growth, with solvable task length doubling approximately every 6–7 months. Code available [1].

1 Introduction

As AI systems are increasingly deployed in open-ended, real-world settings, their capabilities can both exceed and fall short of what is revealed by specific benchmarks or tasks. To make evaluation results meaningful and actionable, we need to align measured model capabilities with human interpretable scales that reflect how people understand task difficulty, competence, and real-world performance.

To quantify AI system capabilities in terms of human capabilities, METR (Kwa et al., 2025) measured AI performance in terms of the length of tasks AI agents can complete and shows exponential growth with capabilities doubling every 7 months. However, this evaluation paradigm relies on expert human completion time annotations, which exhibit high variance between individuals, incur substantial data collection costs, and are impractical to extend consistently across diverse benchmarks.

In this work, we align task difficulty for humans, measured by required completion time, with task difficulty for models, measured by benchmark performance, in order to forecast the real-world capabilities of AI systems. Following prior works (Lalor et al., 2019; Rodriguez et al., 2021; Hofmann et al., 2025), we use the two-parameter logistic (2PL) Item Response Theory (IRT) model to estimate the difficulty of individual tasks and the capabilities of individual models within a unified latent scale. We find that the resulting task difficulty estimates align closely with human completion times. As newer models succeed at solving increasingly difficult and longer-horizon tasks, we estimate that model capabilities double approximately every 6-7 months. These empirical trends are consistent with findings reported by Kwa et al. (2025). Moreover, for newly introduced tasks, our estimated human completion times align well with human annotations and expectations, demonstrating that the proposed IRT based latent scale effectively bridges difficulty estimates between the two entities: humans and models. Overall, our work provides a robust and general framework for evaluating the difficulty of arbitrary tasks and aligning those estimates with human completion time.

2 Related Works

2.1 Measuring capability through human completion time

Estimating task completion time provides a meaningful bridge between benchmarks and real-world applications, enabling researchers to forecast AI capabilities over different time horizons (Ngo, 2023). METR (Kwa et al., 2025) defined the "50%-task-completion time horizon", measuring the human duration of tasks that AI systems can complete with 50% probability and showing that this horizon has grown exponentially over recent years. With human annotators with domain knowledge but no

[1] https://github.com/FY-Liu/BRIDGE

task-specific expertise, they estimated the human duration for a total of 170 tasks.

However, such human annotation is costly to collect, exhibits substantial variance across individuals, and is difficult to extend to new benchmarks or domains. Our work builds on the core insight of grounding evaluation in human difficulty, but replaces direct human time annotation as the primary estimator with a model-based latent difficulty scale. This enables broader coverage while retaining alignment with human completion time.

2.2 Item response theory for large language model benchmarking

Analogous to the evaluation of human abilities in educational testing, psychometric methods such as IRT model have been adapted to language model evaluation to evaluate the difficulty of tasks. Lalor et al. (2019) introduced learning IRT from model generated responses instead of human generated responses to evaluate the difficulty of NLP tasks. Rodriguez et al. (2021) introduced Difficulty and Ability Discriminating (DAD) leaderboards. With IRT, they infer the difficulty, discriminativeness, and feasibility of tasks in the leaderboard. Fluid Benchmarking (Hofmann et al., 2025) applied IRT model to estimate latent model ability and item difficulty, and combines this with adaptive item selection to reduce variance and improve efficiency and validity of benchmarking. It fits an IRT model on a pool of existing LLM responses and uses the learned IRT model to adaptively select questions that fall within the range of model capability.

By mapping performance into a shared latent space, IRT model enables principled comparison between models of different strengths and mitigates benchmark saturation. Our work adopts the IRT foundation, jointly estimate task difficulty and model capability. In addition to prior works which only estimates the latent difficulty, we explicitly anchor the latent difficulty scale to human completion time. This allows IRT derived abilities to be interpreted in real-world, human-centric terms.

2.3 Estimating latent task difficulties

Although fitting IRT models using model outputs requires significantly less effort than human annotators, it still requires a large and diverse set of tasks with sufficient number of attempts. Consequently, a parallel line of research has focused on estimating latent task difficulty more efficiently, with the goal of reducing the number of required attempts. Byrd and Srivastava (2022) used traits correlated with features of questions, answers, and associated contexts, to predict both difficulty and discrimination for new questions. Scarlatos et al. (2025) estimate task difficulty by simulating responses from a population of LLM-based students, each conditioned on a scalar IRT ability parameter so that the model generates answers as if written by learners at different skill levels, automatically scoring these responses, and fitting an Item Response Theory model to infer item difficulty. Truong et al. (2025) trained a model that predicts question difficulty from its embedding features. This line of work is focusing on estimating the latent difficulty of newly added tasks, and can be used in parallel with our work.

2.4 Aligning latent and human task difficulty

TaskSense(Yin et al., 2025) estimates task difficulty in the context of graphical user interface (GUI) tasks by modeling the cognitive processes that precede each user action. It decomposes tasks into sequences of cognitive steps, such as information search, recall, and decision making, and assigns each step a difficulty index based on information-theoretic and cognitive principles. Task difficulty is computed as a linear aggregation of these step-level difficulties. The resulting difficulty estimates are empirically validated against human data and shown to correlate strongly with both step-level and task-level human completion times. Such approach provides rich, task-level interpretations but typically require task-specific modeling and annotations. Our approach differs by treating task difficulty as a latent variable inferred from large-scale model performance across benchmarks, rather than from explicit cognitive decomposition, providing a unified generalizable difficulty estimation framework, aligning with human completion time.

3 Data and Environment

3.1 Data

METR (Kwa et al., 2025) released an aggregated dataset of model performance across a suite of 170 tasks from Software Atomic Actions (SWAA), HCAST (Rein et al., 2025), and RE-Bench (Wijk et al., 2025), with human time annotations for each task. This annotation provides an estimate of "*how long a domain-knowledgeable human without task-specific context*" would take to complete the task.

SWE-bench Verified: SWE-bench (Jimenez et al., 2024) evaluates the software engineering

capabilities of large language models by measuring their ability to resolve real-world GitHub issues. SWE-bench Verified is a curated subset of 500 instances that have been manually validated by human annotators and labeled with task difficulty using four completion time categories: less than 15 minutes, 15 minutes to 1 hour, 1 hour to 4 hours, and more than 4 hours. In our analysis, we use all publicly available leaderboard logs, which report task-level success outcomes for each combination of agent framework and model.

MLE-bench: MLE-bench (Chan et al., 2025) evaluates the capabilities of large language models on end-to-end machine learning workflows, including data preparation, model training, and hyperparameter tuning, etc. The benchmark comprises 75 Kaggle competitions spanning diverse domains and problem types, with tasks categorized into difficulty levels ranging from low to hard. In our experiments, we select a subset of 38 problems, consisting of 20 low difficulty tasks and 18 medium or hard difficulty tasks, due to computational constraints.

GDPval: GDPval (Patwardhan et al., 2025) is a comprehensive benchmark designed to evaluate AI capabilities on economically significant, real-world tasks. It covers 44 distinct occupations across the top 9 sectors contributing to the U.S. GDP, ensuring a broad representation of the economy. The tasks were constructed by industry professionals with an average of 14 years of experience to mirror the actual workflows defined by the U.S. Bureau of Labor Statistics.

3.2 Outcome metrics

METR: METR performed repeated trials for each model-task pair. In this work, we perform a single trial per model-task pair due to limited compute resources and classify it as successful if the model completes the task successfully according to the benchmark-specific success criteria as described in the previous section. Utilizing the METR run logs which consists the binary outcome from a set of models for each task, we treat the model-task pair as successful if the model succeeds at least 50% time over all trials. This choice allows broader coverage across models and tasks.

SWE-bench and MLE-bench: We use verifiable, execution-based outcome metrics to determine whether the model successfully completes a task. In SWE-bench, a task is successfully completed if the model's generated patch resolves the associated GitHub issue as verified by passing all the corresponding unit tests. In MLE-bench, we use three binary indicators of success for each competition: (i) whether the model generates a valid submission under the competition rules (ii) whether this submission achieves performance above median on the public leaderboard and (iii) whether it earns any Kaggle medal (bronze, silver or gold). These 3 indicators are treated as 3 seperate tasks. For each competition, MLE-bench uses Kaggle's public leaderboard for human baselines and discretizes competition scores into these medal tiers for direct comparison with human participants. For both SWE-bench and MLE-bench, we rely on the official evaluation pipelines.

GDPval: There is no automated and verifiable to assess the agent performance on this benchmark due to its open-ended nature and complex workflow. For GDPval, we utilized an automated LLM-as-a-judge framework using Gemini 3 Pro as our judge. This judge applies a custom 1-5 scoring rubric to quantify the quality of the agents' output. A score of 1 corresponds to "almost no work done," a score of 4 denotes "good deliverable text with minor mistakes in deliverable files," and a score of 5 indicates "perfect deliverable text and files," intended to approximate expert-level performance. This rigorous evaluation method allows for a precise measurement of how close current frontier models are to replicating high-value human labor. We treat a score of at least 4 as indicating successful task completion. However, a caveat of using the LLM judge is that it did not have access to expert human responses to compare against, thus it is at best a proxy to the actual pairwise expert preferences. In addition, we also submitted our outputs to OpenAI's public GDPval autograder, but those scores were not yet available at the time of analysis.

3.3 Environment and Models

We conduct all the experiments using the InspectAI agent framework[2] and use the default react agent scaffold with tool access to a bash shell and python interpreter. For each task, generation is capped at 1000 turns or until context window is full. We use the temperature of 0.0 (greedy decoding) and sufficiently large maximum output tokens to avoid truncation of intermediate reasoning steps or final

[2] https://inspect.aisi.org.uk/

outputs. We run each model in an agentic workflow across three benchmarks: SWE-bench, MLE-bench and GDPval. Table 1 in Appendix A lists the different open-source and closed-source models used. We also include the SWE-bench results from the official leaderboard submissions.

4 Methods

Given a set of tasks $T = t_1, ..., t_{|T|}$ and a set of models $M = m_1, ..., m_{|M|}$, for $t_i \in T$ and $m_j \in M$, the probability of having the correct response $P(y_{i,j} = 1)$ is defined by the item response function for the 2PL IRT model as:

$$P(y_{i,j} = 1 \mid \theta_j, a_i, b_i) = \sigma(a_i(\theta_j - b_i) = \frac{e^{a_i(\theta_j - b_i)}}{1 + e^{a_i(\theta_j - b_i)}} \quad (1)$$

Each task t_i is described by its discrimination a_i, how strongly a task distinguishes between models of different ability, and the difficulty b_i. Each model m_j has it's ability estimation θ_j, where σ is the logistic function.

Intuitively, task t_i is not informative when $a_i = 0$, and $a_i \geq 0 \; \forall \; i \in |T|$. Both b_i and θ_j are scale invariant, their absolute values are not identifiable, only their difference $\theta_j - b_i$ matters for the probability of success. Hence b_i and θ_j are centered at zero and are on the same scale. When $P(y_{i,j} = 1 \mid \theta_*, a_i, b_j) = 0.5$, $\theta_* = \beta_i$. θ_* naturally provides the definition for the ability required to achieve 50% success rate on a specific question with difficulty b_i.

The IRT model is fitted to maximize the likelihood $\prod_{i,j} P(y_{ij} \mid \theta_j, a_i, b_i)$ with Markov chain Monte Carlo with hierarchical priors, based on Natesan et al. (2016). We use the implementation from Lalor and Rodriguez (2023).

Using the binary run results of each model-task pair collected from METR, SWE-bench (verified), MLE-bench, and GDPval as described in section 3, we fit the 2PL IRT model and obtained a_i and b_i for each task and θ_j for each model. Although the binary run results are sparse due to limited compute and reproducibility of older frameworks models, the data is still sufficient to fit the 2PL IRT model.

Rewriting eq. (1) in log-odds form: $\log \frac{P}{1-P} = a_i(\theta_j - \beta_i)$. We can measure the capability of model using it's odds $\frac{P}{1-P}$. If the model capability is growing exponentially, i.e. $\frac{P}{1-P} \propto e^{\hat{\theta}}$, $\hat{\theta}$ will grow linearly when it's capability grows exponentially. Being on the same scale as $\hat{\theta}$, $\hat{b}$ will grow linearly when human completion time grows exponentially.

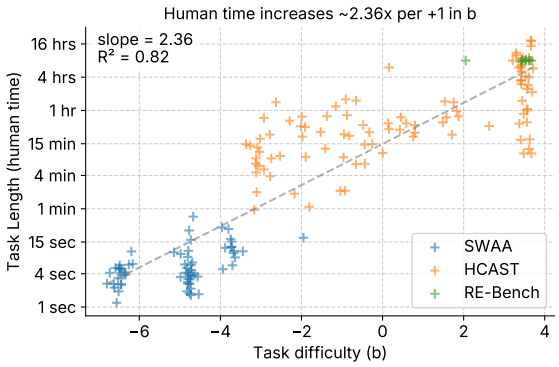

Figure 1: Task length vs. task difficulty on a suite of benchmarks from METR. Task length (human completion time) is shown on a log scale.

Let $T_h \subseteq T$ denote the subset of tasks with human completion time annotations. For each task $t_k \in T_h$, let b_k denote its latent model difficulty and h_k its human completion time. As the model difficulty is on the *log* scale of human completion time, we can fit a linear relationship between $\{b_k : t_k \in T_h\}$ and $\{\log h_k : t_k \in T_h\}$: human time$_i$ $= exp($slope $\times$ b_i + intercept$)$. We can use this relationship to predict the human time for tasks $T_{nh} \not\subset T$ without human time annotation.

5 Experiments and Results

We use the METR aggregated dataset of model-task pairs spanning SWAA, HCAST and RE-Bench with human duration to establish how task difficulty relates to human effort. Specifically, we fit a linear model between the latent task difficulty b and the log of task length (i.e., log of human completion time in minutes). Figure 1 shows that this relationship is well-approximated by a linear function with $R^2 = 0.82$, demonstrating that the latent difficulty scale varies linearly with log human time closely. Moreover, we can also observe that a $+1$ increase in difficulty b corresponds to approx $2.36\times$ increase in log human completion time.

Having established the relationship between latent task difficulty b and task length, we use this mapping to estimate human completion times for three benchmarks that lack exact human time annotations: SWE-bench Verified, MLE-bench, and GDPval. For each task, we take its IRT estimated difficulty b and infer task length using the learned log-linear relationship. Figure 2 presents the resulting task-length distributions as histograms of

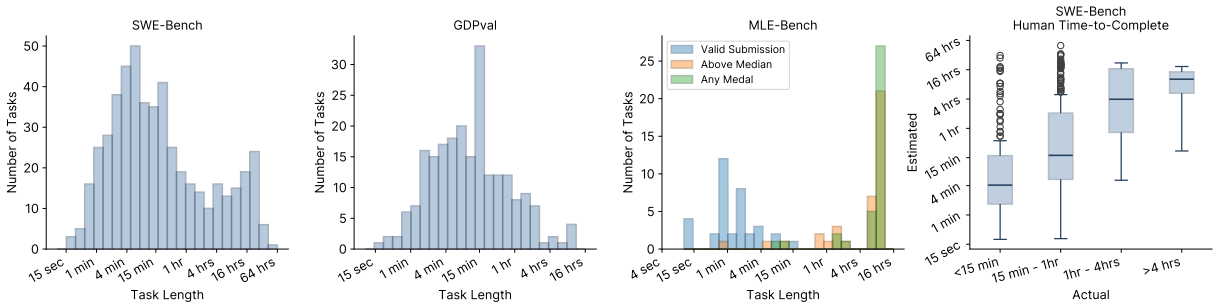

Figure 2: Task length estimation across benchmarks. The first three subfigures on the left show the distribution of predicted task lengths (number of tasks vs. estimated human completion time) for each benchmark. The last plot on the right shows the mean and variance of the predicted time estimates for SWE-bench across different task buckets grouped by the human annotation time.

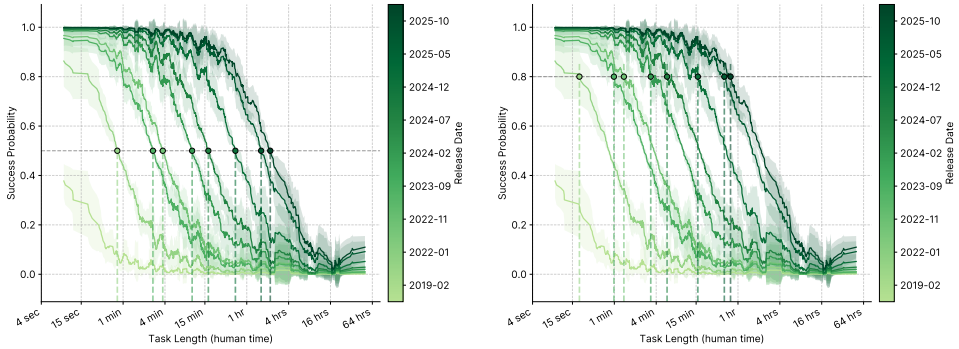

Figure 3: Model success probability versus estimated human task completion time with success probabilities of 50% (left) and 80% (right) across release dates. Task length (human completion time) is estimated using latent difficulty b and is shown on a log scale. Darker green denotes more recent frontier models.

estimated completion time on a log scale.

We validate these estimates on SWE-bench Verified by comparing them against the human-annotated completion-time buckets. The estimated task lengths align well with the annotated buckets for most tasks, with the median estimated time increasing monotonically across buckets. A small number of outliers appear in the shortest-duration buckets, where some tasks are estimated to require longer completion times. We hypothesize that these deviations are largely due to human time annotations provided by repository maintainers who are already familiar with the codebase, rather than by domain experts without task-specific context.

For MLE-bench, tasks corresponding to valid submissions are estimated to require much shorter completion times than those achieving above-median scores or earning medals. Overall, these results indicate that difficulty-based time estimates capture meaningful variation in task horizons and provide a reasonable proxy for human completion time across benchmarks.

To track progress in frontier model capabilities over time, we analyze how task success probability varies as a function of estimated human completion time on a logarithmic scale. We partition models into 5-month release windows, select the model with highest ability θ_{max}within each window, and compute task success probabilities using Equation (1) given θ_{max}, a_i, b_i for all tasks from SWE-bench Verified, MLE-bench, and GDPval.

As shown in Figure 3 (left), success probability exhibits a consistent inverse-sigmoidal relationship with task length: models achieve near perfect performance on short-duration tasks, followed by a sharp decline as task length increases, with performance approaching zero on the longest tasks. Models are ordered by release date, with darker green denoting more recent models. Notably, models released prior to 2022 fail to reach a 50% success rate even on the shortest tasks, underscoring the significant improvement in recent years.

Frontier models released in 2025 achieve 50% success on tasks that would take humans approximately $\sim 2 - 2.5$ hours to complete. However, increasing the success threshold from 50% to 80%

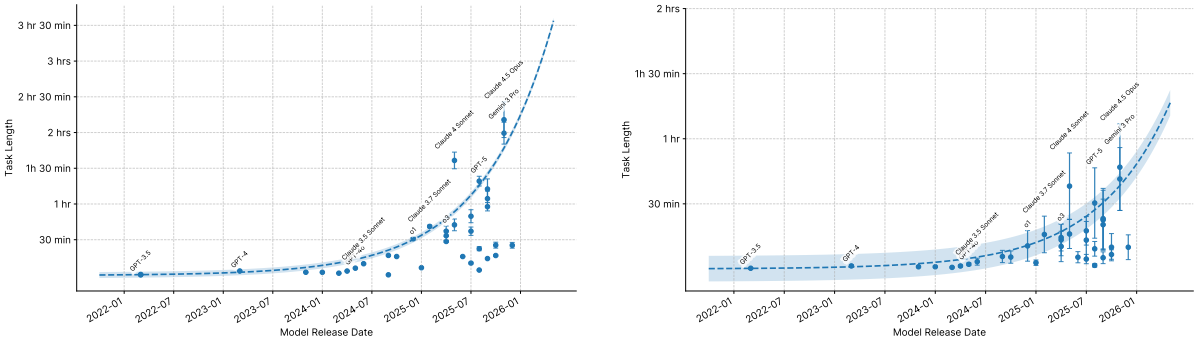

Figure 4: Forecast trends of task length horizon over model release date with success rate of 50% (left) and 80% (right).

substantially reduces the range of solvable task lengths, as shown in Figure 3 (right), where even the most recent models are limited to tasks requiring less than one hour of human effort. We observe analogous trends when analyzing success probability as a function of latent task difficulty b (see Figure 5 in Appendix B).

Finally, we forecast how the task length horizon of frontier models evolves over time. We do so by selecting the best performing models m_s within each 2-month release window and mapping its estimated ability θ_s to the maximum task length solvable at a fixed success rate. At a 50% success rate (Figure 4, left), frontier models exhibit exponential growth in solvable task length, with capabilities doubling approximately every $6-7$ months. The state-of-the-art model reaches a task length horizon of roughly 2 hours at with 50% success rate. We observe analogous trends at an 80% success rate (Figure 4, right), albeit with a reduced achievable horizon, where the state-of-the-art models can solve tasks of just under one hour more reliably. Overall, these forecasts are consistent with prior METR estimates, providing validation that our framework effectively translates latent task difficulty into human completion time horizon, a human interpretable difficulty measurement.

6 Conclusion

We introduced BRIDGE, a framework for aligning model based latent task difficulty with human completion time to enable interpretable and scalable evaluation of AI capabilities. By fitting a 2PL IRT model with binary performance data across diverse benchmarks, we obtain a unified latent scale that captures both task difficulty and model capability. We show that this latent difficulty is strongly and linearly related to the logarithm of human completion time, allowing us to estimate task durations for benchmarks lacking direct human annotations.

Using this alignment, we characterize how model success probability varies with task length and demonstrate that recent frontier models can reliably solve tasks requiring multiple hours of human effort at with 50% success rate. Furthermore, by tracking the evolution of frontier model ability over time, we find that the human task length horizon of state-of-the-art models has grown exponentially, with capabilities doubling roughly every six to seven months, consistent with prior human time-based evaluations.

Team Contributions

Fengyuan: Proposed the BRIDGE idea, implemented IRT with data analysis, writing report.
Jay: Helped brainstorm the BRIDGE idea with Fengyuan, implemented experiments for SWE-bench and MLE-bench, data analysis and visualization, presentation and writing report.
Nilaksh: Implemented experiments with GDPval, LLM as a judge evaluation, tried out different agentic evaluation frameworks, and writing report.
Mentors: Provided guidance and feedback.

Acknowledgements

This research was enabled in part by compute resources provided by Mila (mila.quebec).

References

Matthew Byrd and Shashank Srivastava. 2022. Predicting difficulty and discrimination of natural language questions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*

(Volume 2: Short Papers), pages 119–130, Dublin, Ireland. Association for Computational Linguistics.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Lilian Weng, and Aleksander Mądry. 2025. Mle-bench: Evaluating machine learning agents on machine learning engineering. *arXiv preprint arXiv:2410.07095*.

Valentin Hofmann, David Heineman, Ian Magnusson, Kyle Lo, Jesse Dodge, Maarten Sap, Pang Wei Koh, Chun Wang, Hannaneh Hajishirzi, and Noah A. Smith. 2025. Fluid language model benchmarking. *arXiv preprint arXiv:2509.11106*.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. 2024. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney Von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Harold Miles, Seraphina Nix, Tao Lin, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel M. Ziegler, Elizabeth Barnes, and Lawrence Chan. 2025. Measuring ai ability to complete long tasks. *arXiv preprint arXiv:2503.14499*.

John P. Lalor, Hao Wu, and Hong Yu. 2019. Learning latent parameters without human response patterns: Item response theory with artificial crowds. *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, 2019:4240–4250.

John Patrick Lalor and Pedro Rodriguez. 2023. <tt>py-irt</tt>: A scalable item response theory library for python. *INFORMS Journal on Computing*, 35(1):5–13.

Prathiba Natesan, Ratna Nandakumar, Tom Minka, and Jonathan D Rubright. 2016. Bayesian prior choice in irt estimation using mcmc and variational bayes. *Frontiers in psychology*, 7:1422.

Richard Ngo. 2023. Clarifying and predicting agi.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, Natalie S. Kim, Patrick Chao, Samuel Miserendino, Gildas Chabot, David Li, Michael Sharman, Alexandra Barr, Amelia Glaese, and Jerry Tworek. 2025. Gdpval: Evaluating ai model performance on real-world economically valuable tasks. *arXiv preprint arXiv:2510.04374*.

David Rein, Joel Becker, Amy Deng, Seraphina Nix, Chris Canal, Daniel O'Connel, Pip Arnott, Ryan Bloom, Thomas Broadley, Katharyn Garcia, Brian Goodrich, Max Hasin, Sami Jawhar, Megan Kinniment, Thomas Kwa, Aron Lajko, Nate Rush, Lucas Jun Koba Sato, Sydney Von Arx, Ben West, Lawrence Chan, and Elizabeth Barnes. 2025. Hcast: Human-calibrated autonomy software tasks. *arXiv preprint arXiv:2503.17354*.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P. Lalor, Robin Jia, and Jordan Lee Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change nlp leaderboards? In *Annual Meeting of the Association for Computational Linguistics*.

Alexander Scarlatos, Nigel Fernandez, Christopher Ormerod, Susan Lottridge, and Andrew Lan. 2025. SMART: Simulated students aligned with item response theory for question difficulty prediction. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25082–25105, Suzhou, China. Association for Computational Linguistics.

Sang Truong, Yuheng Tu, Percy Liang, Bo Li, and Sanmi Koyejo. 2025. Reliable and efficient amortized model-based evaluation. *arXiv preprint arXiv:2503.13335*.

Hjalmar Wijk, Tao Lin, Joel Becker, Sami Jawhar, Neev Parikh, Thomas Broadley, Lawrence Chan, Michael Chen, Josh Clymer, Jai Dhyani, Elena Ericheva, Katharyn Garcia, Brian Goodrich, Nikola Jurkovic, Holden Karnofsky, Megan Kinniment, Aron Lajko, Seraphina Nix, Lucas Sato, William Saunders, Maksym Taran, Ben West, and Elizabeth Barnes. 2025. Re-bench: Evaluating frontier ai rd capabilities of language model agents against human experts. *arXiv preprint arXiv:2411.15114*.

Yiwen Yin, Zhian Hu, Xiaoxi Xu, Chun Yu, Xintong Wu, Wenyu Fan, and Yuanchun Shi. 2025. Tasksense: Cognitive chain modeling and difficulty estimation for gui tasks. *arXiv preprint arXiv:2511.09309*.

A Models Used

Provider	Models
Google	`gemini-2.5-flash`, `gemini-2.5-pro`, `gemini-3-pro-preview`
OpenAI	`gpt-4o-mini`, `o3`, `gpt-5.1`, `gpt-5-nano`, `gpt-5-mini`, `gpt-5`, `gpt-oss-20b`, `gpt-oss-120b`
Meta	`meta-llama/Llama-3.3-70B-Instruct`
Qwen	`Qwen/Qwen2.5-Coder-32B-Instruct`, `Qwen/Qwen3-Coder-30B-A3B-Instruct`, `Qwen/Qwen3-Coder-480B-A35B-Instruct`
MiniMax	`MiniMaxAI/MiniMax-M2`

Table 1: List of different models used, grouped by providers.

B Additional Results

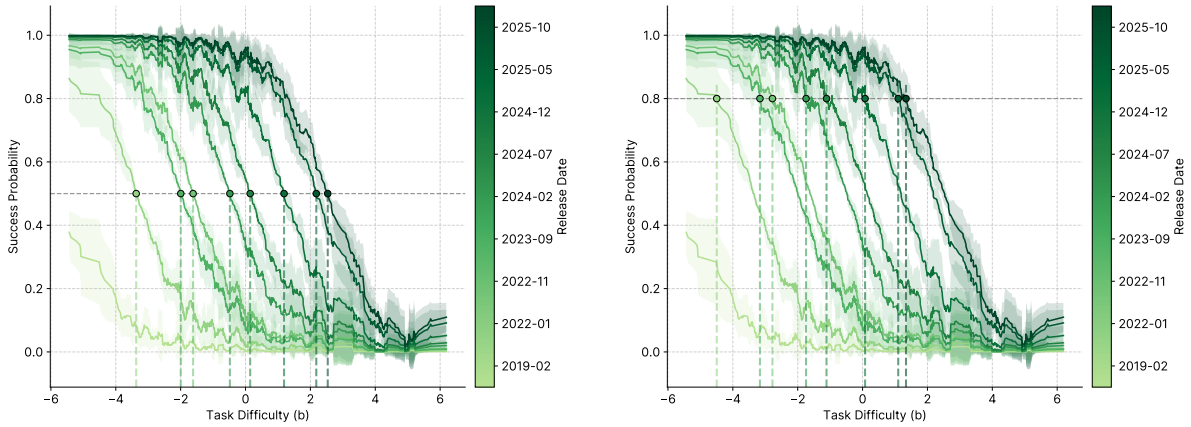

Figure 5: Model success probability versus latent difficulty with success probabilities of 50% (left) and 80% (right) across release dates. Darker green denotes more recent frontier models.