OckBench: Measuring the Efficiency of LLM Reasoning

Zheng Du*[1], Hao Kang*[1], Song Han[2,3], Tushar Krishna[1], and Ligeng Zhu[3]

[1]Georgia Institute of Technology
[2]Massachusetts Institute of Technology
[3]Nvidia

Abstract

Large language models (LLMs) such as GPT-5 and Gemini 3 have pushed the frontier of automated reasoning and code generation. Yet current benchmarks emphasize accuracy and output quality, neglecting a critical dimension: efficiency of token usage. The token efficiency is highly variable in practical. Models solving the same problem with similar accuracy can exhibit up to a **5.0**× difference in token length, leading to massive gap of model reasoning ability. Such variance exposes significant redundancy, highlighting the critical need for a standardized benchmark to quantify the gap of token efficiency. Thus, we introduce OckBench, the first benchmark that jointly measures accuracy and token efficiency across reasoning and coding tasks. Our evaluation reveals that token efficiency remains largely unoptimized across current models, significantly inflating serving costs and latency. These findings provide a concrete roadmap for the community to optimize the latent reasoning ability, token efficiency. Ultimately, we argue for an evaluation paradigm shift: tokens must not be multiplied beyond necessity. Our benchmarks are available at https://ockbench.github.io/.

1 Introduction

"Entities must not be multiplied beyond necessity."

— The Principle of Ockham's Razor

Large Language Models (LLMs) like GPT-5, Gemini 3, and Claude serve as the frontier of automated intelligence, fueled by reasoning techniques like Chain of Thought (CoT) [37]. As the field embraces test-time compute scaling [34], models are increasingly trained to generate extensive token chains to tackle tougher problems. However, this massive inflation of decoding tokens introduces a critical bottleneck. Solving just six problems in the International Olympiad in Informatics can now take over ten hours [26], and complex mathematical challenges (e.g., Putnam math [24]) frequently explode into millions of tokens, turning inference into continuous decoding marathons [23].

While the community celebrates these gains in reasoning capability, prevailing benchmarks like HELM [19] and Chatbot Arena [5] focus almost exclusively on *output quality*, largely ignoring this token efficiency crisis. We argue that this one-dimensional evaluation fails to provide a holistic view of true model intelligence. In reality, many models consume vastly more tokens than necessary to reach the correct answer. As shown in Figure 2a, models of identical size (7B) achieving similar accuracy can differ by over **3.3**× in token consumption and **5.0**× in end-to-end latency.

As models continue to scale and accuracy on standard tasks approaches saturation, relying solely on correctness is no longer sufficient to distinguish capability. This massive variance in computational cost for the exact same correct answer exposes a fundamental blind spot. We propose that efficiency is not merely a logistical constraint, but a proxy for intelligence itself. To comprehensively evaluate a model's true capability and address the pressing bottlenecks in real-world serving, we introduce OckBench: the first benchmark

*Equal contribution. Correspond to zdu@gatech.edu

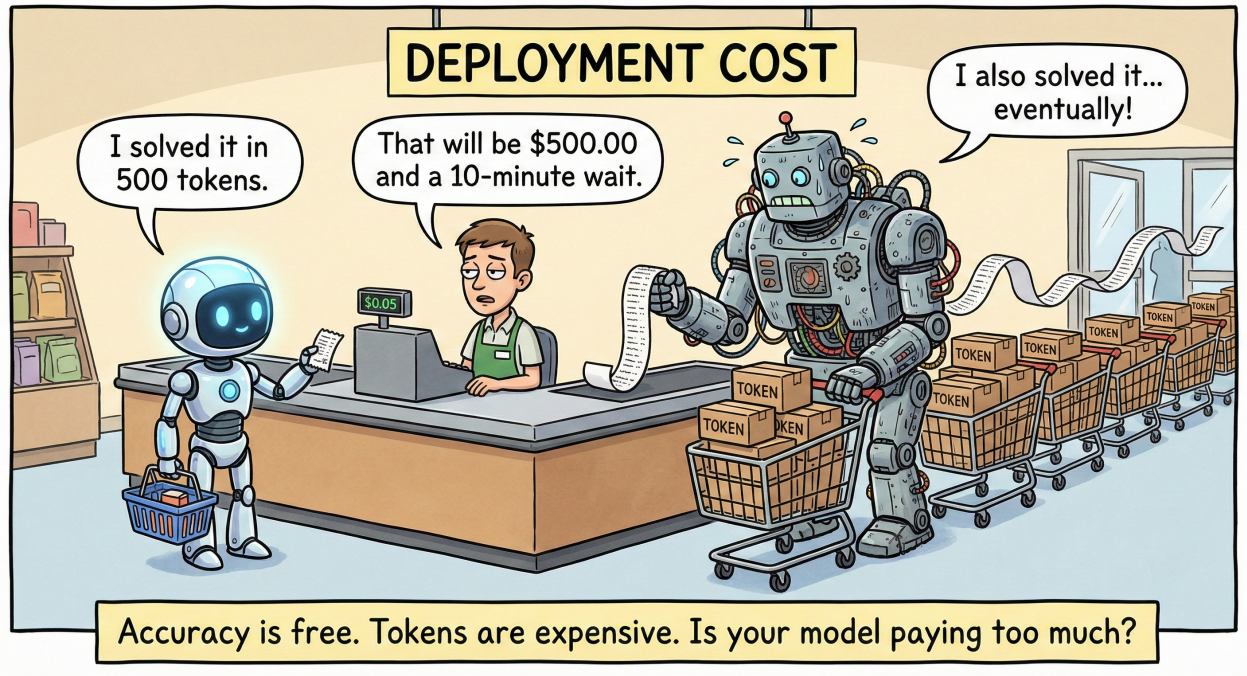


Figure 1: **The Overthinking Tax.** Both agents achieve the same accuracy, yet the verbose model incurs higher latency and cost.

that jointly measures accuracy and token consumption to evaluate deployment-ready reasoning. Through this framework, we formalize the concept of *Per-Token Intelligence*—a critical, yet previously unmeasured dimension that defines a model's ability to solve complex problems not just correctly, but concisely.

To quantify this new dimension of intelligence, we construct OckBench across three core domains: math, coding, and science, drawing from datasets that strictly require reasoning capability, such as AIME, MBPP and GPQA. Rather than evaluating models on randomly sampled questions, OckBench deliberately selects problems of moderate difficulty that naturally elicit a wide variance in token consumption. By focusing on these specific instances, we clearly expose the intrinsic efficiency gap between models that can solve problems concisely and those that resort to verbose babbling. Finally, to provide a holistic ranking, we design the OckScore (S_{Ock}), a unified evaluation metric that logarithmically penalizes unnecessary verbosity while fundamentally prioritizing correctness.

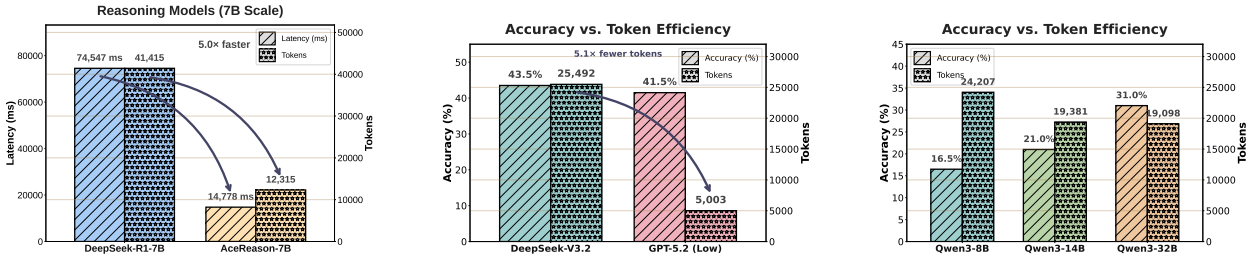
Through extensive evaluations on OckBench, we uncover several critical insights into the current landscape of LLM reasoning. First, we observe an efficiency gap between open-weights and proprietary models. While top-tier open models have nearly closed the performance gap in pure accuracy, they heavily rely on brute-force verbosity. For instance, as shown in Figure 2b, an open-weights model might consume over $\mathbf{5}\times$ the tokens of a closed-source counterpart to solve the exact same problem and achieve comparable accuracy.

Second, we identify a counterintuitive phenomenon within model families: the "Overthinking Tax". While smaller models are generally assumed to be more economical, Figure 2c reveals they often compensate for lower parameter capacity by generating excessively long reasoning chains. Because they consume significantly more tokens to reach the same answer, these ostensibly "cheaper" models paradoxically incur higher latency and total operational costs than their larger counterparts in real-world deployment.

Finally, analyzing the trajectory of frontier models in Figure 3, we observe that newer generations simultaneously improve in both accuracy and token efficiency. This validates our core premise: correctness and efficiency are not isolated metrics, but joint indicators of true intelligence. To meaningfully advance automated reasoning, the community must evaluate and co-optimize both dimensions.

To bridge the gap between pure reasoning accuracy and holistic model intelligence, our core contributions are summarized as follows:

- **The Concept of Per-Token Intelligence:** We introduce a new dimension for evaluating LLMs, proposing that a superior model must not only achieve high accuracy but do so with minimal token



(a) Reasoning Efficiency in 7B Models

(b) Open vs. Closed Source Token Efficiency

(c) The Overthinking Tax in Model Families

Figure 2: **The Efficiency Gap and Overthinking Tax in Reasoning Models.** (a) Despite having the same parameter scale (7B), models can exhibit a 5× difference in latency due to disparate token consumption, emphasizing token efficiency as a critical system bottleneck. (b) While open-weights models (DeepSeek-V3.2) now rival proprietary models (GPT-5.2) in accuracy, they suffer a massive efficiency gap (e.g., consuming 5.1× more tokens), underscoring the necessity of optimizing *Per-Token Intelligence*. (c) Across the Qwen3 series, larger models achieve higher accuracy while consuming *fewer* tokens. Smaller models pay a high "Overthinking Tax" by generating verbose, inefficient reasoning, making them paradoxically more expensive in real-world deployment.

consumption. This paradigm shift provides a concrete roadmap for the community to optimize reasoning capability.

- **OckBench Framework and OckScore:** We present the first hardware-agnostic benchmark that jointly measures accuracy and token efficiency. By employing a novel "Differentiation Filter," OckBench isolates tasks that expose the intrinsic efficiency gap between models. Furthermore, we design the OckScore (S_{Ock}), a unified metric that quantitatively penalizes unnecessary verbosity.
- **Empirical Insights and the "Overthinking Tax":** We systematically evaluate a diverse roster of models, quantifying a severe efficiency gap between open-weights and proprietary architectures. Furthermore, we formally define the "Overthinking Tax," proving that smaller models often incur paradoxically higher deployment costs due to inefficient, verbose reasoning.
- **Validating Optimization Pathways:** We demonstrate that reasoning efficiency is a tractable dimension for optimization. Through training-free model interpolation and difficulty-aware reinforcement learning, we show that models can successfully "Ockhamize" their reasoning and significantly improve their OckScore.

2 Background & Related Work

2.1 The Blind Spot in Current Evaluations

Current benchmarks typically focus on either output quality or system performance, leaving a critical gap.

Accuracy-Centric Benchmarks. Suites like HELM [19] and LM-Eval [12] prioritize correctness. While useful for gauging capability, they treat generation as a black box, ignoring computational costs. Furthermore, preference-based metrics often favor verbose responses, incentivizing models to inflate token counts without adding value – a phenomenon known as *length bias*.

System-Centric Benchmarks. Benchmarks like MLPerf [31] focus on throughput and latency. These measure the *engine* (infrastructure) but not the *driver* (reasoning process). An optimized engine cannot compensate for a model that generates 5× more tokens than necessary.

2.2 The Rising Cost of Reasoning

The need for token-aware evaluation grows with the rise of inference-time compute scaling. As models shift to multi-step reasoning, output lengths explode. Epoch AI reports that reasoning model outputs grow

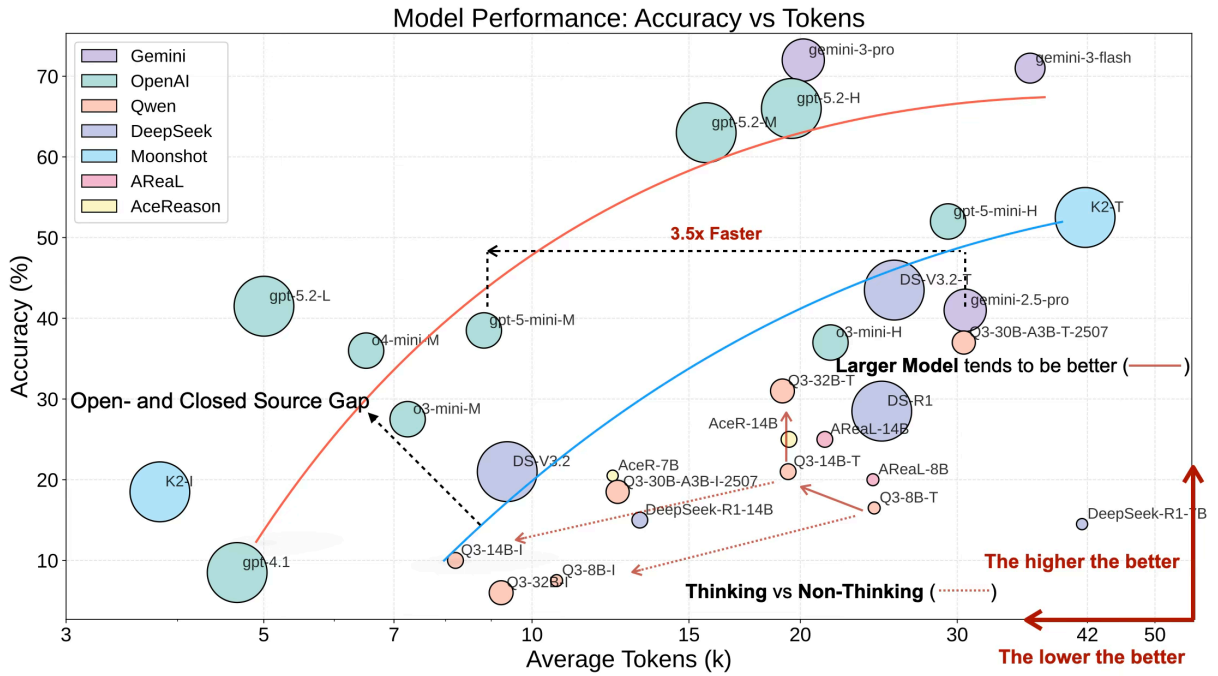


Figure 3: **Accuracy vs. Token Consumption.** A bubble plot illustrating the trade-off between reasoning accuracy (y-axis) and computational cost (x-axis, log scale) across top-tier models. Bubble size indicates model parameter count. The distribution highlights the lack of correlation between pure accuracy and efficiency.

approximately 5× annually, compared to 2.2× for standard models [10].

This trend has practical consequences. Since inference cost scales linearly with token count, providers billing by the token [33] inadvertently penalize efficient reasoning. Without metrics to curb "token bloat," the industry risks optimizing for models that are knowledgeable but operationally intractable.

2.3 Optimization and The "Overthinking" Problem

Existing efficiency research focuses largely on compression (quantization) or system optimizations (FlashAttention) [7, 18, 21], which treat the output distribution as fixed.

Our work targets an orthogonal optimization: **Algorithmic Efficiency.** Recent studies show models often "overthink," generating circular reasoning without improving accuracy [10]. OckBench quantifies this inefficiency, aiding methods like Difficulty-Aware Training (DIET [3]) or model interpolation [38] that aim to boost the *intelligence-per-token* ratio.

3 Benchmark Design: OckBench

OckBench is designed to decouple *reasoning efficiency* from *reasoning capability*. Unlike traditional benchmarks that solely rank models by accuracy, OckBench introduces a selection mechanism that isolates tasks where models exhibit significant divergence in computational cost. This section details our domain composition, our efficiency-sensitive item selection strategy, and our standardized evaluation protocol.

3.1 Benchmark Composition

To ensure broad coverage of reasoning modalities, OckBench aggregates tasks across three complementary domains: mathematics, software engineering, and scientific reasoning.

Mathematics and Reasoning. Mathematical problem-solving serves as the core testbed for logical rigor. We construct our pool from established benchmarks including GSM8K [6], AIME 2024/2025, Olympiad-Bench [15], MATH500 [20], math subset of Humanity's Last Exam [30]and AMO-Bench [1]. To probe frontier reasoning capabilities beyond current saturation points, we additionally incorporate the mathematics subset of Humanity's Last Exam. These datasets cover a spectrum of difficulty from grade-school arithmetic to competition-level number theory.

Software Engineering. Code generation proxies real-world logical synthesis and planning. We utilize a lightweight variant of MBPP [2] and LiveCodeBench [17] to assess practical programming skills.

Scientific Reasoning. We include ScienceQA [22], the STEM subsets of MMLU [16], and GPQA-Diamond [32] to test knowledge-constrained reasoning and concision under technical load.

3.2 Item Selection Strategy: The Differentiation Filter

A naive random sampling of problems is insufficient for benchmarking efficiency. If a problem is too simple, all models solve it concisely (floor effect); if too complex, all models generate long, often incorrect chains (ceiling effect). To ensure OckBench is *sensitive* to intrinsic differences in model efficiency, we employ a *Differentiation Filter*.

Formally, let $\mathcal{M}$ be a diverse set of representative models and $\mathcal{D}_{pool}$ be the initial dataset pool. For each problem $x \in \mathcal{D}_{pool}$, we collect the set of generated token lengths $\mathcal{L}_x = \{\text{len}(m(x)) \mid m \in \mathcal{M}\}$. We select the final benchmark set $\mathcal{D}_{ock}$ based on two criteria:

1. **Difficulty Banding:** We filter for problems where the average accuracy across $\mathcal{M}$ falls within a target band of $0.1 \leq \text{acc}(x) \leq 0.9$. This removes trivial instances and intractable queries, focusing on the "reasoning frontier" where efficiency matters most.

2. **Maximizing Token Variance:** From the remaining pool, we select the top-k instances that maximize the variance of token consumption, $\text{Var}(\mathcal{L}_x)$.

Rationale. High variance in token consumption implies that a problem admits multiple valid reasoning paths—some efficient, some convoluted. By selecting these instances, OckBench highlights the *Efficiency Gap*: the difference between a model that "Ockhamizes" its reasoning and one that "babbles". This ensures that the benchmark penalizes unnecessary verbosity rather than penalizing the necessary complexity required for hard problems.

3.3 Evaluation Protocol

To guarantee reproducibility and fair comparison, we adhere to a strict evaluation protocol.

Inference Settings. We use single-shot prompts and greedy decoding (temperature = 0) to reduce prompt effects and sampling noise.

Correctness Metrics. Accuracy is measured via Pass@1. For mathematics and science tasks, we employ rule-based answer extraction and exact matching against the ground truth. For coding tasks, correctness is verified via functional execution against a suite of unit tests.

Efficiency Metric (Output Tokens). We define "Output Tokens" as the raw count of tokens generated by the model's specific tokenizer prior to the end-of-sequence (EOS) token. Output Tokens include both intermediate reasoning and final answer tokens. While tokenizers differ between architectures, this metric accurately reflects the true deployment cost (latency and compute) specific to each model.

3.4 Unified Score

To design a unified score that can assess both a model's performance on accuracy and efficiency, we propose the *OckScore* (S_{ock}). Current benchmarks typically treat accuracy and computational cost as orthogonal metrics; however, in deployment scenarios, they are inextricably linked. A model that achieves the correct

answer through concise reasoning is strictly superior to one that arrives at the same conclusion through excessive "token babbling," and both are superior to a model that is confident (verbose) but wrong.

Formally, we define our design philosophy through the following preference ordering:

$$\langle \textsc{Correct}, \textsc{Short} \rangle \succ \langle \textsc{Correct}, \textsc{Long} \rangle \succ \langle \textsc{Wrong}, \textsc{Short} \rangle \succ \langle \textsc{Wrong}, \textsc{Long} \rangle$$

To capture this ordinal relationship quantitatively, we introduce a penalized accuracy metric defined as:

$$S_{\text{ock}} = \text{Accuracy} - \lambda \cdot \log\left(\frac{T}{C}\right)$$

where Accuracy is the percentage pass rate (0-100), T is the average number of output tokens, λ is a penalty coefficient, and C is a normalization constant. In our experiments, we set $\lambda = 10$ and $C = 10,000$.

The design of this metric is justified by three key considerations:

Accuracy Prioritization. We fundamentally posit that correctness is the primary utility of a reasoning model. By using accuracy as the base term (0-100) and subtracting a logarithmic penalty, we ensure that efficiency is treated as a *cost* rather than a multiplier. This structure guarantees that a correct solution, no matter how verbose, generally outscores an incorrect one, preserving the boundary between useful and useless models.

Logarithmic Scaling of Efficiency. We employ a logarithmic penalty ($\log T$) rather than a linear one. Reasoning chains vary by orders of magnitude (e.g., 1k vs. 100k tokens). A linear penalty would either be negligible for short chains or disproportionately punitive for long, necessary reasoning. The logarithmic scale compresses this variance, acknowledging that the marginal cost of tokens diminishes in distinctiveness at higher scales, while preventing floor effects where slight verbosity differences in small models dominate the score.

Calibration with Model Priors. The parameters $\lambda = 10$ and $C = 10,000$ were calibrated to align the score with strong empirical priors regarding "per-token intelligence." We operate under the hypothesis that models with higher intelligence (historically, larger commercial models) should optimize the trade-off between accuracy and brevity better than weaker models. Empirically, this parameterization ensures that the score does not falsely promote "short but dumb" models. It yields a ranking where capable commercial models and larger parameter models within the same family (e.g., 32B vs. 8B) score highest, reflecting their ability to condense complex reasoning into efficient paths, whereas smaller or weaker models are penalized for the "overthinking" tax described in Section 5.

4 Experiments

4.1 Experimental Setup and Protocol

Benchmark Composition. To ensure a holistic evaluation of reasoning efficiency, OckBench aggregates tasks across three complementary domains: Mathematics, Software Engineering, and Scientific Reasoning.

- **Mathematics:** We construct our pool from established benchmarks including GSM8K and competition-level datasets such as AIME 2024/2025 and AMO-Bench.
- **Software Engineering:** We utilize variant problems from MBPP and LiveCodeBench to assess practical programming capabilities.
- **Scientific Reasoning:** We incorporate subsets from ScienceQA, MMLU (STEM subjects), and GPQA-Diamond to test reasoning constrained by domain-specific knowledge.

Note: While OckBench covers these three domains, the results and analysis in this section and Section 5 focus primarily on the **Mathematics** subset (OckBench-Math)

Item Selection Strategy. To decouple reasoning efficiency from capability, we employ the *Differentiation Filter* (see Section 3.2). Rather than randomly sampling, we select the top 200 instances exhibiting the greatest variance in token consumption across models. These instances represent the "reasoning frontier" where models diverge most significantly—some solving them concisely, others engaging in excessive "token babbling." This ensures that token count serves as a high-contrast signal for efficiency.

Table 1: **OckBench-Math Leaderboard.** Models are ranked by OckScore (S_{ock}). Gemini 3 Pro achieves the highest efficiency score, balancing high accuracy with moderate token consumption.

Model	Org	#Tokens	Accuracy (%)	Reasoning Efficiency
gemini-3-pro-preview	Gemini	20,154	72.0	67.21
gemini-3-flash-preview	Gemini	36,212	71.0	64.35
gpt-5.2_high	OpenAI	19,541	66.0	61.30
gpt-5.2_medium	OpenAI	15,683	63.0	58.90
gpt-5-mini_high	OpenAI	29,297	52.0	46.06
Kimi-K2-Thinking	Kimi	41,746	52.5	45.36
gpt-5.2_low	OpenAI	5,003	41.5	39.74
DeepSeek-V3.2-Thinking	DeepSeek	25,492	43.5	38.00
o4-mini_high	OpenAI	25,677	43.5	37.98
gemini-2.5-pro	Gemini	30,622	41.0	34.91
Qwen3-235B-A22B-Thinking-2507	Qwen	28,558	38.5	32.64
o3-mini_high	OpenAI	21,623	37.0	32.00
Qwen3-235B-A22B-Instruct-2507	Qwen	7,707	27.0	24.52
DeepSeek-R1	DeepSeek	24,685	28.5	23.10
AReaL-boba-2-32B	AReaL	23,327	27.0	21.77
AceReason-Nemotron-14B	AceReason	19,424	25.0	20.31
DeepSeek-R1-Distill-Qwen-32B	DeepSeek	12,895	15.5	11.90

Evaluation Protocol. All models are evaluated in a single-shot setting to minimize prompt-engineering biases. We employ an LLM-based extraction method where an auxiliary model parses the final answer (e.g., a mathematical value or code block) from the raw response. Correctness is measured via *Pass@1* accuracy. Efficiency is quantified by *Average Output Tokens*, counting the raw tokens generated prior to the end-of-sequence (EOS) token.

4.2 Models

We evaluate a diverse roster of state-of-the-art models, covering both proprietary and open weights across various scales of reasoning effort:

- **Commercial Models:** We evaluate Google's **Gemini 3** [14] and **Gemini 2.5** [8] families. From OpenAI, we evaluate the **GPT-5.2** [28] and **GPT-5-mini** [27] series across three distinct reasoning effort levels (High, Medium, Low), alongside **o3-mini**, **o4-mini** [29], and standard instruction-tuned models like GPT-4o [25].
- **Open-Source Models:** We evaluate a comprehensive suite of open-weights models, led by the **Qwen3** family [36] (including "Thinking" and "Instruct" variants from 4B up to 235B parameters). We also include the **DeepSeek** series, covering the V3.2 and R1 architectures [9, 13], and the **Kimi-K2** series [35]. This is further complemented by the **AReaL-boba-2** series [11] and the **AceReason-Nemotron** models [4].

4.3 Main Results

Table 1 presents the comprehensive results on OckBench-Math, ranking models by our proposed OckScore (S_{Ock}). Figure 3 illustrates the trade-off between accuracy and token consumption. We observe several critical trends:

The Efficiency Gap in Frontier Models. Commercial models dominate the leaderboard yet exhibit stark differences in efficiency. **Gemini-3-Pro** achieves the highest performance with 72.0% accuracy while consuming an average of 20,154 tokens ($S_{Ock} = 67.21$). In contrast, **Gemini-3-Flash** achieves comparable accuracy (71.0%) but requires nearly double the computation (36,212 tokens), resulting in a lower score (64.35). This highlights that distinct models within the same family can possess vastly different "per-token intelligence"—the Flash model effectively compensates for lower parameter capacity by generating significantly longer chains of thought.

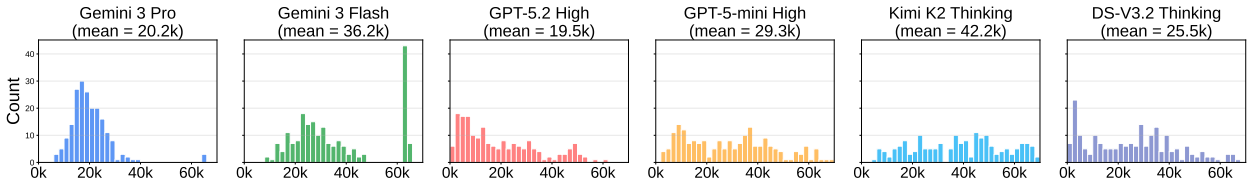


Figure 4: **Token Distribution Across Problems.** Histogram of output-token counts on OckBench-Math for six frontier models (Gemini 3 Pro/Flash, GPT-5.2/5-mini, DeepSeek-V3.2-Thinking and Kimi-K2-Thinking). The distributions expose large variance in reasoning verbosity and reveal saturation behavior for some models near their generation limits.

Kimi-K2 and the Open-Source Trade-off. Among open-weights models, **Kimi-K2-Thinking** emerges as the strongest performer, achieving 52.5% accuracy. Notably, it not only outperforms commercial baselines like GPT-5.2 Low (41.5%) but also rivals **GPT-5-mini High** (52.0%). However, this accuracy comes at a steep computational price: Kimi-K2 consumes an average of 41,746 tokens—the highest in our evaluation and nearly 43% more than GPT-5-mini High (29,297 tokens). This reveals a significant "Efficiency Gap": while top-tier open models have successfully closed the distance in reasoning *accuracy*, they still rely on brute-force, verbose reasoning strategies compared to the more concise paths of commercial models.

Summary. These results validate that accuracy alone is an insufficient metric. Top-performing models distinguish themselves not just by correctness, but by achieving it without the excessive verbosity that plagues less capable or poorly optimized models. While open-source models like Kimi-K2 are reaching commercial-grade accuracy, their lower S_{Ock} scores highlight the urgent need for optimization in reasoning density.

5 Analysis

This section analyzes the token consumption and efficiency of models using OckBench-Math.

5.1 Token Distribution Analysis

Figure 4 summarizes token usage across problems as a distribution, making variance and tail behavior immediately visible. Figure 4 illustrates the distribution of token usage, highlighting variance and tail behaviors. Two patterns emerge: first, output spread varies significantly, with some models exhibiting long, verbose tails. Second, a "saturation" mass appears at the right tail, indicating models frequently hit generation limits where reasoning collapses or loops.

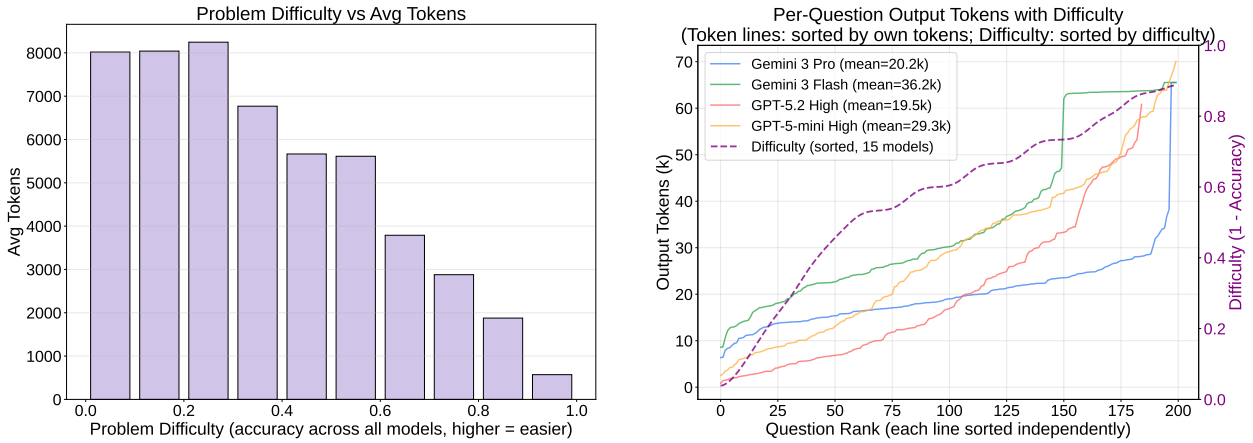
Figure 5b dissects this by problem difficulty, revealing two distinct strategies:

Adaptive vs. Uniform Scaling. The GPT series (GPT-5.2 and GPT-5-mini) exhibits a clear *difficulty-adaptive* trend. As problem difficulty increases (dashed purple line), token consumption rises in a gradual linear or super-linear fashion, suggesting that these models allocate budget proportional to perceived complexity. In contrast, the Gemini family (e.g., Gemini 3 Pro) shows a comparatively *flat* profile over most of the benchmark (ranks 0 to 190), with token output remaining stable (˜20k) and largely insensitive to incremental difficulty.

The "Reasoning Cliff" and Per-Token Intelligence. A critical observation is the "saturation wall" or "cliff" in the Gemini models. Unlike the gradual ramp of the GPT series, Gemini models exhibit a sharp vertical asymptote where token usage spikes abruptly to the model's generation limit (likely the context window cap). In the histogram (Figure 4), this manifests as a distinct frequency spike at the maximum token range (∼64k) for Gemini 3 Flash.

Comparing Gemini 3 Flash and Gemini 3 Pro reveals that this "cliff" is a proxy for model capacity. Gemini 3 Flash hits this saturation point much earlier (around rank 150 in Figure 5b) compared to Pro (rank 195). We argue that the delay of this cliff is a direct indicator of *per-token intelligence*: Gemini 3 Pro

resolves complex reasoning paths more succinctly than Flash, deferring the onset of "reasoning collapse". This validates our hypothesis that efficiency is not only about speed, but about the information density of the generated reasoning chain.



(a) **Reasoning Cost vs. Difficulty.** Analysis across 9 OpenAI models and 3,579 math problems. Average token consumption generally correlates with difficulty but exhibits a resource-intensive "overthinking" tail on the most intractable problems (where accuracy is near zero).

(b) **Token Consumption Profiles.** Comparison of four representative models (Gemini 3 Pro/Flash and GPT-5.2/5-mini). Problems are sorted by output token count for each model. The dashed purple line indicates smoothed problem difficulty, revealing distinct scaling behaviors between model families.

Figure 5: Analysis of reasoning costs and token consumption scaling across different models.

5.2 Tokens vs Difficulty

To quantify how reasoning effort scales with problem complexity, we analyze the relationship between token consumption and problem difficulty, defined here as the average accuracy across all 37 evaluated models. Figure 5a presents the average output tokens for 9 OpenAI reasoning models. We observe a strong negative correlation: as expected, harder problems (lower accuracy) generally necessitate longer reasoning chains. However, a critical inefficiency emerges at the lower bound of accuracy. For intractable problems where models fail to find the correct answer (Accuracy ≈ 0), token consumption does not plateau but rather maintains a high variance. This indicates an "overthinking" behavior, where models expend vast computational resources generating extensive reasoning traces that ultimately fail to converge, effectively paying a high latency cost for zero utility.

5.3 The "Overthinking Tax": When Efficient Models Pay More

One of OckBench's most critical insights is the identification of an *Overthinking Tax*—a phenomenon where models optimized for deployment efficiency (via smaller parameter counts or lower unit pricing) paradoxically incur higher total costs due to excessive verbosity.

Common intuition posits that "smaller is cheaper". To verify this, we conducted a cost analysis using real-world API pricing from SiliconFlow[1]. Comparing DeepSeek-R1-Distill-Qwen-7B and 14B reveals a stark financial inversion. While the 7B model is priced at **50%** of the 14B model per token, it generates 3.13× more output tokens ($41,415$ vs $13,211$) to achieve worse accuracy.

When translating to cost-per-query, the "cheaper" 7B model is effectively **57% more expensive** than the 14B model (14.5 unit cost vs 9.2 unit cost). This demonstrates that aggressive pricing on smaller models is completely negated by the *Overthinking Tax*, rendering them economically inefficient despite their low parameter count. This case demonstrates that efficiency cannot be defined solely by *cost-per-token* or *parameters*. Without the token-aware evaluation provided by OckBench, developers may unknowingly deploy "efficient" models that act as "verbose" bottlenecks, inflating total request latency and operational costs.

[1]Pricing data as of Jan 2026: DeepSeek-R1-Distill-Qwen-7B at \$0.05/1M tokens vs. 14B at \$0.1/1M tokens.

5.4 Efficiency Gap and Trends

Our analysis of the OckBench-Math leaderboard (Figure 3) reveals two critical trends defining the current landscape of reasoning models.

1. The Open-Closed Efficiency Gap. A distinct dichotomy exists between proprietary and open-weights models. While open-source models are closing the accuracy gap, they remain clustered in the lower-right quadrant of the plot—characterized by high token consumption for moderate accuracy. The gap is no longer just about capability, but significantly about *efficiency*. Bridging this divide requires the open-source community to pivot from pure parameter scaling to optimizing reasoning density.

2. Rapid Convergence of Frontier Models. The trajectory of frontier models validates *Per-Token Intelligence* as the primary optimization objective. As illustrated in Figure 3, comparing **gpt-5-mini-medium** with **gemini-2.5-pro** reveals a massive efficiency gap: despite achieving comparable accuracy ($\approx 40\%$), the latter consumed $3.5\times$ more tokens. However, the subsequent generation **gemini-3** has rapidly closed this gap with **gpt-5.2**, shifting towards the Pareto frontier. This swift iteration demonstrates that frontier laboratories are actively optimizing for reasoning efficiency, offering a clear roadmap for open-source development.

6 Improvement Methods

We demonstrate how OckBench verifies efficiency-oriented optimization using two approaches: training-free model interpolation and difficulty-aware reinforcement learning (RL).

6.1 Training-Free Optimization: Model Interpolation

We merge the reasoning of "Thinking" models with the conciseness of "Instruct" models via weight interpolation [38], targeting `Qwen3-Instruct` and `Qwen3-Thinking` checkpoints. Using $\theta_{merged} = 0.6 \cdot \theta_{thinking} + 0.4 \cdot \theta_{instruct}$, we aim to retain reasoning depth while imposing brevity.

Results. The interpolated model (`Qwen3-Mix-6:4`) substantially improves efficiency:

Token Reduction: Average length drops to **12,092 tokens** (from ∼27k).

Accuracy: Relies at **20.0%**, slightly below pure Thinking but above Instruct.

OckScore: Increases to **17.06**.

Interpolation effectively prunes redundant reasoning without retraining.

6.2 Training-Based Optimization: Difficulty-Aware RL

We also employ DIET (Difficulty-Aware Training) [3], an RL framework with dynamic token penalties to mitigate overthinking. We fine-tune `DeepSeek-R1-Distill-Qwen-7B` on DeepScaleR, penalizing length on simple queries while preserving it for complex ones.

Results. Post-training evaluation shows:

Accuracy: Improves to **15.5%** (baseline 14.5%).

Token Efficiency: Average tokens decrease to **23,671** (from 41,415).

OckScore: Jumps to **10.23** (from 7.39).

Table 2 summarizes these results, confirming that reasoning efficiency is a tractable dimension for optimization.

7 Conclusion

In this work, we introduced OckBench, the first model- and hardware-agnostic benchmark designed to measure *Per-Token Intelligence*—the critical trade-off between reasoning accuracy and token consumption. By employing a *Differentiation Filter*, we moved beyond standard accuracy metrics to isolate problems that expose the "Efficiency Gap" between models. We proposed the OckScore (S_{Ock}), a unified metric that penalizes unnecessary verbosity, establishing a new standard for evaluating deployment-ready reasoning.

Our extensive experiments on frontier and open-source models reveal that token efficiency is a significant, yet previously neglected, axis of differentiation. We identified the *Overthinking Tax*, where smaller, distilled models often expend exorbitant computational resources to mimic complex reasoning patterns without

Table 2: **Effectiveness of Efficiency Optimizations.** Comparison of model interpolation (Qwen3-Mix) and difficulty-aware RL (DeepSeek-DIET) against logical baselines on OckBench-Math, showing improvements in OckScore.

Model Strategy	Acc (%)	Avg Tokens	OckScore
Interpolation Method			
Qwen3-4B-Thinking	22.0	27,238	16.29
Qwen3-4B-Instruct	13.5	11,859	10.10
Qwen3-Mix-6:4 (Ours)	**20.0**	**12,092**	**17.06**
RL Optimization Method			
DeepSeek-Distill-Qwen-7B	14.5	41,415	7.39
DeepSeek-DIET (Ours)	**15.5**	**23,671**	**10.23**

converging to correct answers, sometimes incurring higher per-query costs than their larger counterparts despite lower per-token pricing. Furthermore, we demonstrated that efficiency is not a fixed property of model scale but a tractable dimension for optimization. Through model interpolation and Difficulty-Aware Training, we showed that models can be aligned to "Ockhamize" their reasoning, significantly improving their OckScore without retraining from scratch.

We argue that the community must shift from an accuracy-centric evaluation paradigm to one that treats tokens as a cost rather than a free resource. OckBench provides the necessary platform to guide this transition, fostering the development of models that are not only intelligent but also operationally viable. Future work may extend this framework to explore dynamic compute allocation, token-pruning strategies, and efficiency in multimodal reasoning contexts.

References

[1] Shengnan An, Xunliang Cai, Xuezhi Cao, Xiaoyu Li, Yehao Lin, Junlin Liu, Xinxuan Lv, Dan Ma, Xuanlin Wang, Ziwen Wang, and Shuang Zhou. Amo-bench: Large language models still struggle in high school math competitions, 2025. URL https://arxiv.org/abs/2510.26768.

[2] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

[3] Weize Chen, Jiarui Yuan, Tailin Jin, Ning Ding, Huimin Chen, Zhiyuan Liu, and Maosong Sun. The overthinker's diet: Cutting token calories with difficulty-aware training. *arXiv preprint arXiv:2505.19217*, 2025.

[4] Yang Chen, Zhuolin Yang, Zihan Liu, Chankyu Lee, Peng Xu, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Acereason-nemotron: Advancing math and code reasoning through reinforcement learning. *arXiv preprint arXiv:2505.16400*, 2025.

[5] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL https://arxiv.org/abs/2403.04132.

[6] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

[7] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022. URL https://arxiv.org/abs/2205.14135.

[8] Google DeepMind. Introducing gemini 2.5 pro: Advanced reasoning model, 2025. URL https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-pro. Accessed: 2025-10-31.

[9] DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Erhang Li, Fangqi Zhou, Fangyun Lin, Fucong Dai, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoran Wei, Haowei Zhang, Haowen Luo, Haozhe Ji, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, Jialiang Huang, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jingchang Chen, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jinhua Zhu, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexin Huang, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Liang Zhao, Liangsheng Yin, Lihua Guo, Lingxiao Luo, Linwang Ma, Litong Wang, Liyue Zhang, M. S. Di, M. Y Xu, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Panpan Huang, Peixin Cong, Peiyi Wang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, S. H. Liu, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Songyang Zhou, Tao Ni, Tao Yun, Tian Pei, Tian Ye, Tianyuan Yue, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjun Gao, Wentao Zhang, Xi Gao, Xiangwen Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyuan Li, Xu Chen, Xuecheng Su, Xuehai Pan, Xuheng Lin, Xuwei Fu, Y. Q. Wang, Yang Zhang, Yanhong Xu, Yanru Ma, Yao Li, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yiliang Xiong, Ying He, Ying Zhou, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuduan Wang, Yue Gong, Yuhan Wu, Yuheng Zou, Yukun Li, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehua Zhao, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhiyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Zizheng Pan, Zongqing Yao, Bei Feng, Hui Li, J. L. Cai, Jiaqi Ni, Lei Xu, Meng Li, Ning Tian, R. J. Chen, R. L. Jin, S. S. Li, Shuang Zhou, Tianyu Sun, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xinnan Song, Xinyi Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, Dongjie Ji, Jian Liang, Jianzhong Guo, Jin Chen, Leyi Xia, Miaojun Wang, Mingming Li, Peng Zhang, Ruyi Chen, Shangmian Sun, Shaoqing Wu, Shengfeng Ye, T. Wang, W. L. Xiao, Wei An, Xianzu Wang, Xiaowen Sun, Xiaoxiang Wang, Ying Tang, Yukun Zha, Zekai Zhang, Zhe Ju, Zhen Zhang, and Zihua Qu. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

[10] Luke Emberson, Ben Cottier, Josh You, Tom Adamczewski, and Jean-Stanislas Denain. Llm responses to benchmark questions are getting longer over time, 2025. URL https://epoch.ai/data-insights/output-length. Accessed: 2025-10-19.

[11] Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. Areal: A large-scale asynchronous reinforcement learning system for language reasoning, 2025. URL https://arxiv.org/abs/2505.24298.

[12] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

[13] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[14] Demis Hassabis, Koray Kavukcuoglu, and the Gemini team. A new era of intelligence with gemini

3. https://blog.google/products-and-platforms/products/gemini/gemini-3/, November 2025. Accessed 2026-01-29.

[15] Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024. URL https://arxiv.org/abs/2402.14008.

[16] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

[17] Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

[18] Hao Kang, Qingru Zhang, Souvik Kundu, Geonhwa Jeong, Zaoxing Liu, Tushar Krishna, and Tuo Zhao. Gear: An efficient kv cache compression recipe for near-lossless generative inference of llm, 2024. URL https://arxiv.org/abs/2403.05527.

[19] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=iO4LZibEqW. Featured Certification, Expert Certification.

[20] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step, 2023. URL https://arxiv.org/abs/2305.20050.

[21] Yujun Lin, Haotian Tang, Shang Yang, Zhekai Zhang, Guangxuan Xiao, Chuang Gan, and Song Han. Qserve: W4a8kv4 quantization and system co-design for efficient llm serving, 2025. URL https://arxiv.org/abs/2405.04532.

[22] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[23] Thang Luong, Dawsen Hwang, Hoang H. Nguyen, Golnaz Ghiasi, Yuri Chervonyi, Insuk Seo, Junsu Kim, Garrett Bingham, Jonathan Lee, Swaroop Mishra, Alex Zhai, Clara Huiyi Hu, Henryk Michalewski, Jimin Kim, Jeonghyun Ahn, Junhwi Bae, Xingyou Song, Trieu H. Trinh, Quoc V. Le, and Junehyuk Jung. Towards robust mathematical reasoning, 2025. URL https://arxiv.org/abs/2511.01846.

[24] Mathematical Association of America. Maa putnam: William lowell putnam mathematical competition. https://maa.org/putnam/. Accessed 2026-02-21.

[25] OpenAI. Hello gpt-4o ("o" for "omni"), 2024. URL https://openai.com/index/hello-gpt-4o/. Accessed: 2025-10-31.

[26] OpenAI. Learning to reason with llms, September 2024. URL https://openai.com/index/learning-to-reason-with-llms/.

[27] OpenAI. Introducing gpt-5, 2025. URL https://openai.com/index/introducing-gpt-5/. Accessed: 2025-10-31.

[28] OpenAI. Introducing gpt-5.2, 2025. URL https://openai.com/index/introducing-gpt-5-2/. Accessed: 2026-01-29.

[29] OpenAI. Introducing openai o3 and o4 mini, 2025. URL https://openai.com/index/introducing-o3-and-o4-mini/. Accessed: 2026-01-29.

[30] Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, Adam Khoja, Ryan Kim, Richard Ren, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Dmitry Dodonov, Tung Nguyen, Jaeho Lee, Daron Anderson, Mikhail Doroshenko, Alun Cennyth Stokes, Mobeen Mahmood, Oleksandr Pokutnyi, Oleg Iskra, Jessica P. Wang, John-Clark Levin, Mstyslav Kazakov, Fiona Feng, Steven Y. Feng, Haoran Zhao, Michael Yu, Varun Gangal, Chelsea Zou, Zihan Wang, Serguei Popov, Robert Gerbicz, Geoff Galgon, Johannes Schmitt, Will Yeadon, Yongki Lee, Scott Sauers, Alvaro Sanchez, Fabian Giska, Marc Roth, Søren Riis, Saiteja Utpala, Noah Burns, Gashaw M. Goshu, Mohinder Maheshbhai Naiya, Chidozie Agu, Zachary Giboney, Antrell Cheatom, Francesco Fournier-Facio, Sarah-Jane Crowson, Lennart Finke, Zerui Cheng, Jennifer Zampese, Ryan G. Hoerr, Mark Nandor, Hyunwoo Park, Tim Gehrunger, Jiaqi Cai, Ben McCarty, Alexis C Garretson, Edwin Taylor, Damien Sileo, Qiuyu Ren, Usman Qazi, Lianghui Li, Jungbae Nam, John B. Wydallis, Pavel Arkhipov, Jack Wei Lun Shi, Aras Bacho, Chris G. Willcocks, Hangrui Cao, Sumeet Motwani, Emily de Oliveira Santos, Johannes Veith, Edward Vendrow, Doru Cojoc, Kengo Zenitani, Joshua Robinson, Longke Tang, Yuqi Li, Joshua Vendrow, Natanael Wildner Fraga, Vladyslav Kuchkin, Andrey Pupasov Maksimov, Pierre Marion, Denis Efremov, Jayson Lynch, Kaiqu Liang, Aleksandar Mikov, Andrew Gritsevskiy, Julien Guillod, Gözdenur Demir, Dakotah Martinez, Ben Pageler, Kevin Zhou, Saeed Soori, Ori Press, Henry Tang, Paolo Rissone, Sean R. Green, Lina Brüssel, Moon Twayana, Aymeric Dieuleveut, Joseph Marvin Imperial, Ameya Prabhu, Jinzhou Yang, Nick Crispino, Arun Rao, Dimitri Zvonkine, Gabriel Loiseau, Mikhail Kalinin, Marco Lukas, Ciprian Manolescu, Nate Stambaugh, Subrata Mishra, Tad Hogg, Carlo Bosio, Brian P Coppola, Julian Salazar, Jaehyeok Jin, Rafael Sayous, Stefan Ivanov, Philippe Schwaller, Shaipranesh Senthilkuma, Andres M Bran, Andres Algaba, Kelsey Van den Houte, Lynn Van Der Sypt, Brecht Verbeken, David Noever, Alexei Kopylov, Benjamin Myklebust, Bikun Li, Lisa Schut, Evgenii Zheltonozhskii, Qiaochu Yuan, Derek Lim, Richard Stanley, Tong Yang, John Maar, Julian Wykowski, Martí Oller, Anmol Sahu, Cesare Giulio Ardito, Yuzheng Hu, Ariel Ghislain Kemogne Kamdoum, Alvin Jin, Tobias Garcia Vilchis, Yuexuan Zu, Martin Lackner, James Koppel, Gongbo Sun, Daniil S. Antonenko, Steffi Chern, Bingchen Zhao, Pierrot Arsene, Joseph M Cavanagh, Daofeng Li, Jiawei Shen, Donato Crisostomi, Wenjin Zhang, Ali Dehghan, Sergey Ivanov, David Perrella, Nurdin Kaparov, Allen Zang, Ilia Sucholutsky, Arina Kharlamova, Daniil Orel, Vladislav Poritski, Shalev Ben-David, Zachary Berger, Parker Whitfill, Michael Foster, Daniel Munro, Linh Ho, Shankar Sivarajan, Dan Bar Hava, Aleksey Kuchkin, David Holmes, Alexandra Rodriguez-Romero, Frank Sommerhage, Anji Zhang, Richard Moat, Keith Schneider, Zakayo Kazibwe, Don Clarke, Dae Hyun Kim, Felipe Meneguitti Dias, Sara Fish, Veit Elser, Tobias Kreiman, Victor Efren Guadarrama Vilchis, Immo Klose, Ujjwala Anantheswaran, Adam Zweiger, Kaivalya Rawal, Jeffery Li, Jeremy Nguyen, Nicolas Daans, Haline Heidinger, Maksim Radionov, Václav Rozhoň, Vincent Ginis, Christian Stump, Niv Cohen, Rafał Poświata, Josef Tkadlec, Alan Goldfarb, Chenguang Wang, Piotr Padlewski, Stanislaw Barzowski, Kyle Montgomery, Ryan Stendall, Jamie Tucker-Foltz, Jack Stade, T. Ryan Rogers, Tom Goertzen, Declan Grabb, Abhishek Shukla, Alan Givré, John Arnold Ambay, Archan Sen, Muhammad Fayez Aziz, Mark H Inlow, Hao He, Ling Zhang, Younesse Kaddar, Ivar Ängquist, Yanxu Chen, Harrison K Wang, Kalyan Ramakrishnan, Elliott Thornley, Antonio Terpin, Hailey Schoelkopf, Eric Zheng, Avishy Carmi, Ethan D. L. Brown, Kelin Zhu, Max Bartolo, Richard Wheeler, Martin Stehberger, Peter Bradshaw, JP Heimonen, Kaustubh Sridhar, Ido Akov, Jennifer Sandlin, Yury Makarychev, Joanna Tam, Hieu Hoang, David M. Cunningham, Vladimir Goryachev, Demosthenes Patramanis, Michael Krause, Andrew Redenti, David Aldous, Jesyin Lai, Shannon Coleman, Jiangnan Xu, Sangwon Lee, Ilias Magoulas, Sandy Zhao, Ning Tang, Michael K. Cohen, Orr Paradise, Jan Hendrik Kirchner, Maksym Ovchynnikov, Jason O. Matos, Adithya Shenoy, Michael Wang, Yuzhou Nie, Anna Sztyber-Betley, Paolo Faraboschi, Robin Riblet, Jonathan Crozier, Shiv Halasyamani, Shreyas Verma, Prashant Joshi, Eli Meril, Ziqiao Ma, Jérémy Andréoletti, Raghav Singhal, Jacob Platnick, Volodymyr Nevirkovets, Luke Basler, Alexander Ivanov, Seri Khoury, Nils Gustafsson, Marco Piccardo, Hamid Mostaghimi, Qijia Chen,

Virendra Singh, Tran Quoc Khánh, Paul Rosu, Hannah Szlyk, Zachary Brown, Himanshu Narayan, Aline Menezes, Jonathan Roberts, William Alley, Kunyang Sun, Arkil Patel, Max Lamparth, Anka Reuel, Linwei Xin, Hanmeng Xu, Jacob Loader, Freddie Martin, Zixuan Wang, Andrea Achilleos, Thomas Preu, Tomek Korbak, Ida Bosio, Fereshteh Kazemi, Ziye Chen, Biró Bálint, Eve J. Y. Lo, Jiaqi Wang, Maria Inês S. Nunes, Jeremiah Milbauer, M Saiful Bari, Zihao Wang, Behzad Ansarinejad, Yewen Sun, Stephane Durand, Hossam Elgnainy, Guillaume Douville, Daniel Tordera, George Balabanian, Hew Wolff, Lynna Kvistad, Hsiaoyun Milliron, Ahmad Sakor, Murat Eron, Andrew Favre D. O., Shailesh Shah, Xiaoxiang Zhou, Firuz Kamalov, Sherwin Abdoli, Tim Santens, Shaul Barkan, Allison Tee, Robin Zhang, Alessandro Tomasiello, G. Bruno De Luca, Shi-Zhuo Looi, Vinh-Kha Le, Noam Kolt, Jiayi Pan, Emma Rodman, Jacob Drori, Carl J Fossum, Niklas Muennighoff, Milind Jagota, Ronak Pradeep, Honglu Fan, Jonathan Eicher, Michael Chen, Kushal Thaman, William Merrill, Moritz Firsching, Carter Harris, Stefan Ciobâcă, Jason Gross, Rohan Pandey, Ilya Gusev, Adam Jones, Shashank Agnihotri, Pavel Zhelnov, Mohammadreza Mofayezi, Alexander Piperski, David K. Zhang, Kostiantyn Dobarskyi, Roman Leventov, Ignat Soroko, Joshua Duersch, Vage Taamazyan, Andrew Ho, Wenjie Ma, William Held, Ruicheng Xian, Armel Randy Zebaze, Mohanad Mohamed, Julian Noah Leser, Michelle X Yuan, Laila Yacar, Johannes Lengler, Katarzyna Olszewska, Claudio Di Fratta, Edson Oliveira, Joseph W. Jackson, Andy Zou, Muthu Chidambaram, Timothy Manik, Hector Haffenden, Dashiell Stander, Ali Dasouqi, Alexander Shen, Bita Golshani, David Stap, Egor Kretov, Mikalai Uzhou, Alina Borisovna Zhidkovskaya, Nick Winter, Miguel Orbegozo Rodriguez, Robert Lauff, Dustin Wehr, Colin Tang, Zaki Hossain, Shaun Phillips, Fortuna Samuele, Fredrik Ekström, Angela Hammon, Oam Patel, Faraz Farhidi, George Medley, Forough Mohammadzadeh, Madellene Peñaflor, Haile Kassahun, Alena Friedrich, Rayner Hernandez Perez, Daniel Pyda, Taom Sakal, Omkar Dhamane, Ali Khajegili Mirabadi, Eric Hallman, Kenchi Okutsu, Mike Battaglia, Mohammad Maghsoudimehrabani, Alon Amit, Dave Hulbert, Roberto Pereira, Simon Weber, Handoko, Anton Peristyy, Stephen Malina, Mustafa Mehkary, Rami Aly, Frank Reidegeld, Anna-Katharina Dick, Cary Friday, Mukhwinder Singh, Hassan Shapourian, Wanyoung Kim, Mariana Costa, Hubeyb Gurdogan, Harsh Kumar, Chiara Ceconello, Chao Zhuang, Haon Park, Micah Carroll, Andrew R. Tawfeek, Stefan Steinerberger, Daattavya Aggarwal, Michael Kirchhof, Linjie Dai, Evan Kim, Johan Ferret, Jainam Shah, Yuzhou Wang, Minghao Yan, Krzysztof Burdzy, Lixin Zhang, Antonio Franca, Diana T. Pham, Kang Yong Loh, Joshua Robinson, Abram Jackson, Paolo Giordano, Philipp Petersen, Adrian Cosma, Jesus Colino, Colin White, Jacob Votava, Vladimir Vinnikov, Ethan Delaney, Petr Spelda, Vit Stritecky, Syed M. Shahid, Jean-Christophe Mourrat, Lavr Vetoshkin, Koen Sponselee, Renas Bacho, Zheng-Xin Yong, Florencia de la Rosa, Nathan Cho, Xiuyu Li, Guillaume Malod, Orion Weller, Guglielmo Albani, Leon Lang, Julien Laurendeau, Dmitry Kazakov, Fatimah Adesanya, Julien Portier, Lawrence Hollom, Victor Souza, Yuchen Anna Zhou, Julien Degorre, Yiğit Yalın, Gbenga Daniel Obikoya, Rai, Filippo Bigi, M. C. Boscá, Oleg Shumar, Kaniuar Bacho, Gabriel Recchia, Mara Popescu, Nikita Shulga, Ngefor Mildred Tanwie, Thomas C. H. Lux, Ben Rank, Colin Ni, Matthew Brooks, Alesia Yakimchyk, Huanxu, Liu, Stefano Cavalleri, Olle Häggström, Emil Verkama, Joshua Newbould, Hans Gundlach, Leonor Brito-Santana, Brian Amaro, Vivek Vajipey, Rynaa Grover, Ting Wang, Yosi Kratish, Wen-Ding Li, Sivakanth Gopi, Andrea Caciolai, Christian Schroeder de Witt, Pablo Hernández-Cámara, Emanuele Rodolà, Jules Robins, Dominic Williamson, Vincent Cheng, Brad Raynor, Hao Qi, Ben Segev, Jingxuan Fan, Sarah Martinson, Erik Y. Wang, Kaylie Hausknecht, Michael P. Brenner, Mao Mao, Christoph Demian, Peyman Kassani, Xinyu Zhang, David Avagian, Eshawn Jessica Scipio, Alon Ragoler, Justin Tan, Blake Sims, Rebeka Plecnik, Aaron Kirtland, Omer Faruk Bodur, D. P. Shinde, Yan Carlos Leyva Labrador, Zahra Adoul, Mohamed Zekry, Ali Karakoc, Tania C. B. Santos, Samir Shamseldeen, Loukmane Karim, Anna Liakhovitskaia, Nate Resman, Nicholas Farina, Juan Carlos Gonzalez, Gabe Maayan, Earth Anderson, Rodrigo De Oliveira Pena, Elizabeth Kelley, Hodjat Mariji, Rasoul Pouriamanesh, Wentao Wu, Ross Finocchio, Ismail Alarab, Joshua Cole, Danyelle Ferreira, Bryan Johnson, Mohammad Safdari, Liangti Dai, Siriphan Arthornthurasuk, Isaac C. McAlister, Alejandro José Moyano, Alexey Pronin, Jing Fan, Angel Ramirez-Trinidad, Yana Malysheva, Daphiny Pottmaier, Omid Taheri, Stanley Stepanic, Samuel Perry, Luke Askew, Raúl Adrián Huerta Rodríguez, Ali M. R. Minissi, Ricardo Lorena, Krishnamurthy Iyer, Arshad Anil Fasiludeen, Ronald Clark, Josh Ducey, Matheus Piza, Maja Somrak, Eric Vergo, Juehang Qin, Benjámin Borbás, Eric Chu, Jack Lindsey, Antoine Jallon, I. M. J. McInnis, Evan Chen, Avi Semler, Luk Gloor, Tej Shah, Marc Carauleanu, Pascal Lauer, Tran Duc Huy, Hossein Shahrtash, Emilien Duc, Lukas Lewark, Assaf Brown, Samuel Albanie,

Brian Weber, Warren S. Vaz, Pierre Clavier, Yiyang Fan, Gabriel Poesia Reis e Silva, Long, Lian, Marcus Abramovitch, Xi Jiang, Sandra Mendoza, Murat Islam, Juan Gonzalez, Vasilios Mavroudis, Justin Xu, Pawan Kumar, Laxman Prasad Goswami, Daniel Bugas, Nasser Heydari, Ferenc Jeanplong, Thorben Jansen, Antonella Pinto, Archimedes Apronti, Abdallah Galal, Ng Ze-An, Ankit Singh, Tong Jiang, Joan of Arc Xavier, Kanu Priya Agarwal, Mohammed Berkani, Gang Zhang, Zhehang Du, Benedito Alves de Oliveira Junior, Dmitry Malishev, Nicolas Remy, Taylor D. Hartman, Tim Tarver, Stephen Mensah, Gautier Abou Loume, Wiktor Morak, Farzad Habibi, Sarah Hoback, Will Cai, Javier Gimenez, Roselynn Grace Montecillo, Jakub Łucki, Russell Campbell, Asankhaya Sharma, Khalida Meer, Shreen Gul, Daniel Espinosa Gonzalez, Xavier Alapont, Alex Hoover, Gunjan Chhablani, Freddie Vargus, Arunim Agarwal, Yibo Jiang, Deepakkumar Patil, David Outevsky, Kevin Joseph Scaria, Rajat Maheshwari, Abdelkader Dendane, Priti Shukla, Ashley Cartwright, Sergei Bogdanov, Niels Mündler, Sören Möller, Luca Arnaboldi, Kunvar Thaman, Muhammad Rehan Siddiqi, Prajvi Saxena, Himanshu Gupta, Tony Fruhauff, Glen Sherman, Mátyás Vincze, Siranut Usawasutsakorn, Dylan Ler, Anil Radhakrishnan, Innocent Enyekwe, Sk Md Salauddin, Jiang Muzhen, Aleksandr Maksapetyan, Vivien Rossbach, Chris Harjadi, Mohsen Bahaloohoreh, Claire Sparrow, Jasdeep Sidhu, Sam Ali, Song Bian, John Lai, Eric Singer, Justine Leon Uro, Greg Bateman, Mohamed Sayed, Ahmed Menshawy, Darling Duclosel, Dario Bezzi, Yashaswini Jain, Ashley Aaron, Murat Tiryakioglu, Sheeshram Siddh, Keith Krenek, Imad Ali Shah, Jun Jin, Scott Creighton, Denis Peskoff, Zienab EL-Wasif, Ragavendran P V, Michael Richmond, Joseph McGowan, Tejal Patwardhan, Hao-Yu Sun, Ting Sun, Nikola Zubić, Samuele Sala, Stephen Ebert, Jean Kaddour, Manuel Schottdorf, Dianzhuo Wang, Gerol Petruzella, Alex Meiburg, Tilen Medved, Ali ElSheikh, S Ashwin Hebbar, Lorenzo Vaquero, Xianjun Yang, Jason Poulos, Vilém Zouhar, Sergey Bogdanik, Mingfang Zhang, Jorge Sanz-Ros, David Anugraha, Yinwei Dai, Anh N. Nhu, Xue Wang, Ali Anil Demircali, Zhibai Jia, Yuyin Zhou, Juncheng Wu, Mike He, Nitin Chandok, Aarush Sinha, Gaoxiang Luo, Long Le, Mickaël Noyé, Michał Perełkiewicz, Ioannis Pantidis, Tianbo Qi, Soham Sachin Purohit, Letitia Parcalabescu, Thai-Hoa Nguyen, Genta Indra Winata, Edoardo M. Ponti, Hanchen Li, Kaustubh Dhole, Jongee Park, Dario Abbondanza, Yuanli Wang, Anupam Nayak, Diogo M. Caetano, Antonio A. W. L. Wong, Maria del Rio-Chanona, Dániel Kondor, Pieter Francois, Ed Chalstrey, Jakob Zsambok, Dan Hoyer, Jenny Reddish, Jakob Hauser, Francisco-Javier Rodrigo-Ginés, Suchandra Datta, Maxwell Shepherd, Thom Kamphuis, Qizheng Zhang, Hyunjun Kim, Ruiji Sun, Jianzhu Yao, Franck Dernoncourt, Satyapriya Krishna, Sina Rismanchian, Bonan Pu, Francesco Pinto, Yingheng Wang, Kumar Shridhar, Kalon J. Overholt, Glib Briia, Hieu Nguyen, David, Soler Bartomeu, Tony CY Pang, Adam Wecker, Yifan Xiong, Fanfei Li, Lukas S. Huber, Joshua Jaeger, Romano De Maddalena, Xing Han Lù, Yuhui Zhang, Claas Beger, Patrick Tser Jern Kon, Sean Li, Vivek Sanker, Ming Yin, Yihao Liang, Xinlu Zhang, Ankit Agrawal, Li S. Yifei, Zechen Zhang, Mu Cai, Yasin Sonmez, Costin Cozianu, Changhao Li, Alex Slen, Shoubin Yu, Hyun Kyu Park, Gabriele Sarti, Marcin Briański, Alessandro Stolfo, Truong An Nguyen, Mike Zhang, Yotam Perlitz, Jose Hernandez-Orallo, Runjia Li, Amin Shabani, Felix Juefei-Xu, Shikhar Dhingra, Orr Zohar, My Chiffon Nguyen, Alexander Pondaven, Abdurrahim Yilmaz, Xuandong Zhao, Chuanyang Jin, Muyan Jiang, Stefan Todoran, Xinyao Han, Jules Kreuer, Brian Rabern, Anna Plassart, Martino Maggetti, Luther Yap, Robert Geirhos, Jonathon Kean, Dingsu Wang, Sina Mollaei, Chenkai Sun, Yifan Yin, Shiqi Wang, Rui Li, Yaowen Chang, Anjiang Wei, Alice Bizeul, Xiaohan Wang, Alexandre Oliveira Arrais, Kushin Mukherjee, Jorge Chamorro-Padial, Jiachen Liu, Xingyu Qu, Junyi Guan, Adam Bouyamourn, Shuyu Wu, Martyna Plomecka, Junda Chen, Mengze Tang, Jiaqi Deng, Shreyas Subramanian, Haocheng Xi, Haoxuan Chen, Weizhi Zhang, Yinuo Ren, Haoqin Tu, Sejong Kim, Yushun Chen, Sara Vera Marjanović, Junwoo Ha, Grzegorz Luczyna, Jeff J. Ma, Zewen Shen, Dawn Song, Cedegao E. Zhang, Zhun Wang, Gaël Gendron, Yunze Xiao, Leo Smucker, Erica Weng, Kwok Hao Lee, Zhe Ye, Stefano Ermon, Ignacio D. Lopez-Miguel, Theo Knights, Anthony Gitter, Namkyu Park, Boyi Wei, Hongzheng Chen, Kunal Pai, Ahmed Elkhanany, Han Lin, Philipp D. Siedler, Jichao Fang, Ritwik Mishra, Károly Zsolnai-Fehér, Xilin Jiang, Shadab Khan, Jun Yuan, Rishab Kumar Jain, Xi Lin, Mike Peterson, Zhe Wang, Aditya Malusare, Maosen Tang, Isha Gupta, Ivan Fosin, Timothy Kang, Barbara Dworakowska, Kazuki Matsumoto, Guangyao Zheng, Gerben Sewuster, Jorge Pretel Villanueva, Ivan Rannev, Igor Chernyavsky, Jiale Chen, Deepayan Banik, Ben Racz, Wenchao Dong, Jianxin Wang, Laila Bashmal, Duarte V. Gonçalves, Wei Hu, Kaushik Bar, Ondrej Bohdal, Atharv Singh Patlan, Shehzaad Dhuliawala, Caroline Geirhos, Julien Wist, Yuval Kansal, Bingsen Chen, Kutay Tire, Atak Talay Yücel, Brandon Christof, Veerupaksh Singla, Zijian Song, Sanxing Chen, Jiaxin Ge, Kaustubh Ponkshe, Isaac

Park, Tianneng Shi, Martin Q. Ma, Joshua Mak, Sherwin Lai, Antoine Moulin, Zhuo Cheng, Zhanda Zhu, Ziyi Zhang, Vaidehi Patil, Ketan Jha, Qiutong Men, Jiaxuan Wu, Tianchi Zhang, Bruno Hebling Vieira, Alham Fikri Aji, Jae-Won Chung, Mohammed Mahfoud, Ha Thi Hoang, Marc Sperzel, Wei Hao, Kristof Meding, Sihan Xu, Vassilis Kostakos, Davide Manini, Yueying Liu, Christopher Toukmaji, Jay Paek, Eunmi Yu, Arif Engin Demircali, Zhiyi Sun, Ivan Dewerpe, Hongsen Qin, Roman Pflugfelder, James Bailey, Johnathan Morris, Ville Heilala, Sybille Rosset, Zishun Yu, Peter E. Chen, Woongyeong Yeo, Eeshaan Jain, Ryan Yang, Sreekar Chigurupati, Julia Chernyavsky, Sai Prajwal Reddy, Subhashini Venugopalan, Hunar Batra, Core Francisco Park, Hieu Tran, Guilherme Maximiano, Genghan Zhang, Yizhuo Liang, Hu Shiyu, Rongwu Xu, Rui Pan, Siddharth Suresh, Ziqi Liu, Samaksh Gulati, Songyang Zhang, Peter Turchin, Christopher W. Bartlett, Christopher R. Scotese, Phuong M. Cao, Ben Wu, Jacek Karwowski, Davide Scaramuzza, Aakaash Nattanmai, Gordon McKellips, Anish Cheraku, Asim Suhail, Ethan Luo, Marvin Deng, Jason Luo, Ashley Zhang, Kavin Jindel, Jay Paek, Kasper Halevy, Allen Baranov, Michael Liu, Advaith Avadhanam, David Zhang, Vincent Cheng, Brad Ma, Evan Fu, Liam Do, Joshua Lass, Hubert Yang, Surya Sunkari, Vishruth Bharath, Violet Ai, James Leung, Rishit Agrawal, Alan Zhou, Kevin Chen, Tejas Kalpathi, Ziqi Xu, Gavin Wang, Tyler Xiao, Erik Maung, Sam Lee, Ryan Yang, Roy Yue, Ben Zhao, Julia Yoon, Sunny Sun, Aryan Singh, Ethan Luo, Clark Peng, Tyler Osbey, Taozhi Wang, Daryl Echeazu, Hubert Yang, Timothy Wu, Spandan Patel, Vidhi Kulkarni, Vijaykaarti Sundarapandiyan, Ashley Zhang, Andrew Le, Zafir Nasim, Srikar Yalam, Ritesh Kasamsetty, Soham Samal, Hubert Yang, David Sun, Nihar Shah, Abhijeet Saha, Alex Zhang, Leon Nguyen, Laasya Nagumalli, Kaixin Wang, Alan Zhou, Aidan Wu, Jason Luo, Anwith Telluri, Summer Yue, Alexandr Wang, and Dan Hendrycks. Humanity's last exam, 2025. URL https://arxiv.org/abs/2501.14249.

[31] Vijay Janapa Reddi, Christine Cheng, David Kanter, Peter Mattson, Guenther Schmuelling, Carole-Jean Wu, Brian Anderson, Maximilien Breughe, Mark Charlebois, William Chou, Ramesh Chukka, Cody Coleman, Sam Davis, Pan Deng, Greg Diamos, Jared Duke, Dave Fick, J. Scott Gardner, Itay Hubara, Sachin Idgunji, Thomas B. Jablin, Jeff Jiao, Tom St. John, Pankaj Kanwar, David Lee, Jeffery Liao, Anton Lokhmotov, Francisco Massa, Peng Meng, Paulius Micikevicius, Colin Osborne, Gennady Pekhimenko, Arun Tejusve Raghunath Rajan, Dilip Sequeira, Ashish Sirasao, Fei Sun, Hanlin Tang, Michael Thomson, Frank Wei, Ephrem Wu, Lingjie Xu, Koichi Yamada, Bing Yu, George Yuan, Aaron Zhong, Peizhao Zhang, and Yuchen Zhou. Mlperf inference benchmark, 2019.

[32] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023. URL https://arxiv.org/abs/2311.12022.

[33] Rohail Saleem. Duolingo allegedly tops a list of openai's top 30 customers by token consumption, October 2025. URL https://wccftech.com/duolingo-allegedly-tops-a-list-of-openais-top-30-customers-by-token-consumption/.

[34] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters, 2024. URL https://arxiv.org/abs/2408.03314.

[35] Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Hao Hu, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Chao Hong, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Lijun Lu, Shengling Ma, Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Xinjie Sun, Flood Sung, Heyi Tang, Jiawen Tao, Qifeng Teng, Chensi Wang, Dinglu Wang, Feng Wang, Haiming Wang, Jianzhou Wang, Jiaxing Wang, Jinhong Wang, Shengjie Wang, Shuyi Wang, Yao Wang, Yejie Wang, Yiqin Wang, Yuxin Wang, Yuzhi

Wang, Zhaoji Wang, Zhengtao Wang, Zhexu Wang, Chu Wei, Qianqian Wei, Wenhao Wu, Xingzhe Wu, Yuxin Wu, Chenjun Xiao, Xiaotong Xie, Weimin Xiong, Boyu Xu, Jing Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ziyao Xu, Junjie Yan, Yuzi Yan, Xiaofei Yang, Ying Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Xingcheng Yao, Wenjie Ye, Zhuorui Ye, Bohong Yin, Longhui Yu, Enming Yuan, Hongbang Yuan, Mengjie Yuan, Haobing Zhan, Dehao Zhang, Hao Zhang, Wanlu Zhang, Xiaobin Zhang, Yangkun Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Haotian Zhao, Yikai Zhao, Huabin Zheng, Shaojie Zheng, Jianren Zhou, Xinyu Zhou, Zaida Zhou, Zhen Zhu, Weiyu Zhuang, and Xinxing Zu. Kimi k2: Open agentic intelligence, 2025. URL https://arxiv.org/abs/2507.20534.

[36] Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

[37] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

[38] Taiqiang Wu, Runming Yang, Tao Liu, Jiahao Wang, and Ngai Wong. Revisiting model interpolation for efficient reasoning. *arXiv preprint arXiv:2510.10977*, 2025.