Unified Deployment-Aware Evaluation of Open Reasoning Language Models

Anonymous authors
Paper under double-blind review

Abstract

Open reasoning language models are often compared under mixed sample sizes, partially standardized prompts, and accuracy-centered summaries, which makes practical model selection difficult to interpret. We present a unified evaluation of seven open reasoning language model configurations across four benchmarks, namely ARC-Challenge, GSM8K, MATH levels 1 to 3, and TruthfulQA MC1, under three prompting strategies: zero-shot, chain-of-thought (CoT), and few-shot CoT. Every model–dataset–strategy condition is evaluated on the same 238-example subset, which yields a complete $7 \times 4 \times 3$ design with 84 conditions and 19,992 evaluated examples. In addition to accuracy, we report Wilson confidence intervals, latency, peak video random access memory (VRAM), weighted aggregate performance, Pareto-efficient operating points, prompt-sensitivity metrics, and compatibility diagnostics. Under this unified protocol, the highest weighted score is achieved by Gemma-4-26B-A4B with zero-shot prompting at 0.794, while Gemma-4-E4B remains close to the top across prompting settings with substantially lower latency and memory, making it a particularly attractive practical operating point. Bootstrap and paired-permutation analyses show that top weighted configurations are close enough that deployment tradeoffs remain important. We further find that prompting strategy changes ranking order rather than simply shifting all models in the same direction, and that benchmark-specific complementarity creates measurable routing headroom: an oracle task-aware selector reaches a weighted score of 0.825. Finally, compatibility diagnostics reveal that some apparent failures, especially for Phi-4-Reasoning on GSM8K, reflect deployment-relevant robustness and interface-adherence problems under the shared evaluation pipeline. These results support a central claim: open-model evaluation should be framed as a deployment-aware, multi-objective operating-point problem rather than as a single-score leaderboard exercise.

1 Introduction

Open large language models (LLMs) are increasingly evaluated not only by raw benchmark accuracy, but also by robustness, efficiency, and deployment cost. Recent evaluation frameworks and surveys have argued that reliable LLM assessment requires standardized protocols, broader metric coverage, and explicit attention to inference efficiency rather than accuracy alone Liang et al. (2023); Laskar et al. (2024); Chang et al. (2024). This is especially important for reasoning-oriented models, whose observed performance can change substantially under different prompting strategies such as zero-shot prompting, chain-of-thought (CoT) prompting, and few-shot CoT prompting Wei et al. (2022); Kojima et al. (2022); Wang et al. (2023); Sclar et al. (2024).

At the same time, recent work has highlighted that evaluation conclusions can be sensitive to seemingly minor methodological choices. Prompt formatting alone can produce large performance swings in open models, which raises concerns about comparing models under arbitrarily chosen prompt templates Sclar et al. (2024). More broadly, recent reviews of LLM evaluation have emphasized that inconsistent dataset choices, heterogeneous protocols, and incomplete reporting often make cross-paper conclusions difficult to compare or reproduce Laskar et al. (2024); Chang et al. (2024). These concerns are particularly relevant in reasoning evaluation, where benchmark results are often used to support strong claims about model capability.

This paper argues that the problem is not only incomplete standardization, but also incomplete framing. In many practical settings, model choice is not a pure leaderboard problem. A configuration with the highest aggregate score may be too slow, too memory-intensive, too prompt-sensitive, or too task-specific to be the most attractive deployment choice. Conversely, a configuration that is not the absolute score leader may offer a substantially better operating point once latency, video random access memory (VRAM), prompting stability, and task-dependent behavior are considered together. Our aim is therefore not merely to compare several open reasoning models, but to show that open-model evaluation should move from single-score ranking toward deployment-aware operating-point analysis.

We study that problem directly through a unified evaluation of seven open reasoning language model configurations on four widely used benchmark tasks: ARC-Challenge Clark et al. (2018), GSM8K Cobbe et al. (2021), MATH levels 1 to 3 Hendrycks et al. (2021), and TruthfulQA MC1 Lin et al. (2022). Each model is tested under three prompting strategies: zero-shot, CoT, and few-shot CoT. Crucially, all 84 model–dataset–strategy conditions are evaluated under the same per-condition sample size of 238 examples. The resulting design is complete, balanced, and directly comparable.

The purpose of the study is not to argue that one current model family is universally best. Instead, the goal is to understand what becomes visible when open reasoning models are evaluated under a fully unified and deployment-aware protocol. We focus on the following research questions:

1. How do rankings change when all model–dataset–strategy conditions are evaluated under a single matched-size protocol?
2. Does the highest weighted-score configuration coincide with the most attractive practical operating point under latency and memory constraints?
3. To what extent does prompting strategy alter ranking order across models?
4. Which task-specific patterns remain stable across model families under unified evaluation?
5. How much headroom is suggested by cross-task complementarity under an oracle task-aware routing upper bound?

The main contributions of the paper are as follows.

1. We present a fully unified benchmark protocol for open reasoning language models that fixes the sample size, prompting families, extraction pipeline, and deployment-oriented reporting across all evaluated conditions.
2. We show that the highest weighted-score configuration and the strongest practical operating point are not necessarily the same, and we formalize that distinction through Pareto-frontier analysis, deployment-budget summaries, and resource-normalized efficiency metrics.
3. We show that prompting strategy changes ranking order across models rather than acting as a simple uniform performance booster, and we quantify this instability through cross-strategy rank-correlation and prompt-sensitivity analyses.
4. We show that benchmark-specific complementarity creates measurable routing headroom: an oracle task-aware selector achieves a higher weighted score than the best single fixed configuration, which motivates future deployment-aware routing without making routing the main contribution of the present paper.
5. We provide compatibility and failure diagnostics showing that some apparent benchmark failures, especially for Phi-4-Reasoning under GSM8K and MATH-style settings, reflect deployment-relevant interface robustness problems under the shared evaluation pipeline.

Taken together, these contributions support a central claim: open reasoning model evaluation should be treated as a multi-objective operating-point selection problem rather than as a single-score leaderboard exercise.

The rest of the paper is organized as follows. Section 2 reviews prior work on reasoning benchmarks, prompting-based reasoning, prompt sensitivity, and holistic LLM evaluation. Section 3 describes the unified methodology in detail. Section 4 reports the main empirical results, including deployment-aware operating points, oracle routing headroom, and compatibility diagnostics. Section 5 discusses what these results imply for model evaluation and deployment, and also presents limitations and future work. Section 6 concludes the paper.

2 Related Work

2.1 Reasoning Benchmarks for Large Language Models

A substantial body of recent work evaluates LLMs on benchmarks designed to probe reasoning rather than shallow pattern matching. ARC-Challenge was introduced as a science question answering benchmark specifically constructed to be difficult for retrieval and co-occurrence baselines, thereby emphasizing more substantive reasoning ability Clark et al. (2018). GSM8K has become a standard benchmark for grade-school mathematical reasoning and multi-step arithmetic problem solving Cobbe et al. (2021). MATH extends this line of evaluation to competition-level mathematics and was designed to test mathematical problem-solving ability at a substantially higher level of difficulty Hendrycks et al. (2021). TruthfulQA provides a complementary perspective by measuring whether models reproduce common human falsehoods and misconceptions, rather than merely maximizing answer plausibility Lin et al. (2022). Taken together, these datasets cover distinct but complementary aspects of reasoning, factual reliability, and answer validity.

2.2 Prompting-Based Reasoning and Prompt Sensitivity

Prompting strategy has become central to reasoning evaluation. Chain-of-thought prompting showed that providing intermediate reasoning demonstrations can substantially improve model performance on arithmetic, commonsense, and symbolic reasoning tasks Wei et al. (2022). Zero-shot CoT later demonstrated that explicit reasoning can also be elicited without exemplars through simple reasoning-trigger phrases, suggesting that prompt design itself is a major determinant of observed capability Kojima et al. (2022). Self-consistency further showed that decoding strategy interacts with CoT prompting in important ways, yielding additional gains by aggregating across multiple sampled reasoning paths Wang et al. (2023).

More recent work has shown that prompt sensitivity is not merely a nuisance variable but a serious methodological issue. Sclar et al. show that open LLMs can be highly sensitive to meaning-preserving prompt formatting changes in few-shot settings, and argue that reporting a single prompt format can mischaracterize model quality Sclar et al. (2024). Related robustness work has similarly found that LLMs are vulnerable to prompt perturbations and adversarial prompt variations, reinforcing the need to treat prompting choices as an experimental factor rather than a minor implementation detail Zhu et al. (2023); Gan & Mori (2023).

2.3 Holistic and Standardized LLM Evaluation

A parallel line of work argues that LLM evaluation should be broader, denser, and more standardized. HELM is especially influential in this regard, because it explicitly promotes scenario coverage, multi-metric reporting, and standardized comparison across models Liang et al. (2023). Recent survey papers further document that heterogeneous evaluation setups, incomplete reporting, and protocol mismatch remain widespread problems in the literature Chang et al. (2024); Laskar et al. (2024). In the open-model setting, recent leaderboard efforts have also attempted to improve comparability and reduce evaluation artifacts. For example, the Open-LLM-Leaderboard paper argues that multiple-choice evaluation can hide issues such as selection bias and random guessing, and proposes open-style evaluation to better reflect model capability Myrzakhan et al. (2024). Tooling work such as LLMBox likewise emphasizes unified interfaces for training, inference, and evaluation, reflecting the field's broader shift toward reproducible and systematized benchmarking Tang et al. (2024).

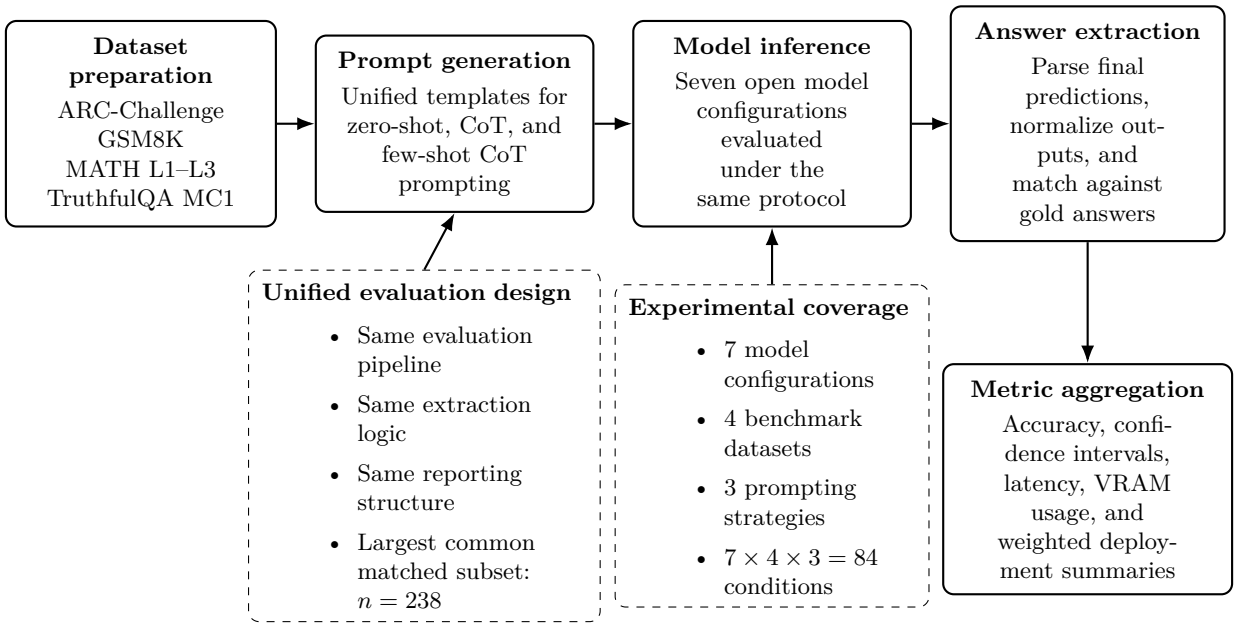


Figure 1: Overview of the unified evaluation workflow. The study uses a single standardized pipeline for dataset preparation, prompt construction, model inference, output extraction, and metric aggregation across all model-dataset-strategy conditions. The protocol evaluates seven open model configurations on four benchmarks under three prompting strategies, yielding 84 directly comparable conditions, each evaluated on the largest common matched subset of $n = 238$

2.4 Position of the Present Study

The present study is closest in spirit to holistic and standardized benchmarking, but it makes a narrower methodological intervention. Rather than proposing a new benchmark, we ask what can be learned from evaluating existing open reasoning models under a strictly unified protocol with matched sample size, fixed prompting families, and deployment-oriented measurements. In that sense, the paper is not only a comparison of model families; it is also an empirical study of how unified evaluation design affects ranking, interpretation, and deployment-relevant conclusions.

3 Methodology

We evaluate seven open reasoning language model configurations across four datasets and three prompting strategies under a fully unified protocol. Figure 1 illustrates the workflow. The pipeline contains five main stages: dataset preparation, prompt generation, model inference, answer extraction, and metric aggregation.

3.1 Benchmark tasks

The unified evaluation uses four benchmark datasets: ARC-Challenge for multiple-choice scientific reasoning Clark et al. (2018), GSM8K for grade-school mathematical reasoning Cobbe et al. (2021), MATH for mathematical problem solving Hendrycks et al. (2021), and TruthfulQA MC1 for multiple-choice truthfulness evaluation Lin et al. (2022). These benchmarks were chosen to cover complementary evaluation dimensions: ARC-Challenge emphasizes science-oriented reasoning under a multiple-choice format, GSM8K emphasizes multi-step grade-school mathematical reasoning, MATH L1–L3 emphasizes more advanced mathematical problem solving, and TruthfulQA MC1 emphasizes truthfulness and resistance to plausible but misleading answers.

For all four benchmarks, evaluation was conducted on the project's prepared unified subset, with 238 examples per model–dataset–strategy condition. This matched-size design was chosen to ensure direct comparabil-

ity across all 84 conditions. In the project configuration, all four prepared benchmark subsets were generated with `sample_size = 238` and `seed = 42`. The preparation pipeline selected examples by applying a seed-controlled random shuffle and then saving the selected index set to the corresponding `data/indices/*.csv` file for reproducibility (publicly archived at: `https://anonymous.4open.science/r/UDAE-D371/`). For `MATH L1-L3`, the preparation pipeline first filtered the benchmark to levels 1–3, yielding a filtered subset of 238 examples under the current setup; to preserve a fully matched protocol, the other three benchmarks were also prepared at 238 examples. We therefore use 238 not as a claim of an optimal or power-derived sample size, but as the largest common matched subset available under the current unified preparation pipeline. The resulting experiment matrix contains $7 \times 4 \times 3 = 84$ complete conditions and 19,992 evaluated examples.

3.2 Model configurations

The evaluation includes seven open model configurations:

- Gemma-4-26B-A4B
- Gemma-4-E2B
- Gemma-4-E4B
- Phi-4-Mini-Reasoning
- Phi-4-Reasoning
- Qwen3-30B-A3B
- Qwen3-8B

The current project configuration records the following architecture and parameter metadata. Phi-4-Mini-Reasoning is a dense model with 3.8B total parameters and 3.8B active parameters. Gemma-4-E2B is a mixture-of-experts (MoE) model with 5.0B total parameters and 2.0B active parameters. Gemma-4-E4B is an MoE model with 8.0B total parameters and 4.0B active parameters. Qwen3-30B-A3B is an MoE model with 30.0B total parameters and 3.0B active parameters. Gemma-4-26B-A4B is an MoE model with 26.0B total parameters and 3.8B active parameters. Qwen3-8B is a dense model with 8.0B total parameters and 8.0B active parameters. Phi-4-Reasoning is a dense model with 14.0B total parameters and 14.0B active parameters.

Official release references for the evaluated model families include the Gemma 4 release materials for Gemma-4-E2B, Gemma-4-E4B, and Gemma-4-26B-A4B Google (2026b;a), the Phi-4-Reasoning technical report and Phi-4-Mini-Reasoning model documentation Abdin et al. (2025); Microsoft (2025), and the official Qwen3 release materials and model documentation for Qwen3-30B-A3B and Qwen3-8B Qwen Team (2025c;a;b).

3.3 Prompting strategies

Each model was evaluated under three prompting strategies: `zero-shot`, chain-of-thought (CoT), and few-shot chain-of-thought (few-shot CoT). This design treats prompting as a first-class experimental factor rather than as a minor implementation detail. We therefore do not assume that one prompting method is uniformly best across all model families and benchmark tasks. Instead, we study whether the relative ordering of models changes across prompt conditions.

The prompt family was intentionally kept minimal and uniform across datasets and model families. This was a methodological choice rather than an attempt to reproduce each benchmark's original default prompt format. Because prompt sensitivity is itself a central concern in both the prior literature and the present study, we use a shared prompt scaffold to reduce benchmark-specific prompt engineering and to isolate how ranking behavior changes under a common evaluation interface. The goal is therefore controlled comparability, not prompt optimization for any single benchmark.

The exact prompt construction procedure for zero-shot, CoT, and few-shot CoT evaluation is documented in Appendix A. The corresponding benchmark code, configuration files, and curated evaluation summaries are publicly archived at `https://anonymous.4open.science/r/UDAE-D371/`.

3.4 Inference protocol and implementation

All runs were executed through a common benchmark pipeline under a unified evaluation protocol. For each model–dataset–strategy condition, the system recorded binary correctness, per-example latency, output token count, token throughput, and peak video random access memory (VRAM) usage. Condition-level accuracy was computed as the mean of the binary correctness field, and Wilson confidence intervals were reported for all condition-level accuracy estimates.

Experiments were conducted on the `axis2` server, which is equipped with eight NVIDIA H100 80GB high-bandwidth memory 3 (HBM3) graphics processing units (GPUs), each with 81,559 MiB of memory. The software environment used Python 3.13.11, PyTorch 2.10.0 compiled with CUDA 12.8, `transformers` 5.6.2, `datasets` 4.8.4, and `accelerate` 1.13.0. CUDA was available, and all benchmark runs were executed with GPU-backed inference rather than central processing unit (CPU) fallback.

Inference was executed through the shared runner `experiments/run_benchmark_with_fallback. py`. This runner loads prepared records from `data/prepared/`, constructs prompts using the shared prompt builder, optionally applies the tokenizer chat template, performs generation, extracts the final answer, and grades the prediction against the gold label. All model configurations listed in `configs/models.yaml` were marked with `use_chat_template: true`, so prompts were passed through the tokenizer chat-template interface uniformly across models. The default precision in the model configuration was `bfloat16`. The loader attempted to initialize each model first in `bf16` mode and then, when needed, in 4-bit mode through `BitsAndBytesConfig`. Because the extracted environment report did not confirm a working `bitsandbytes` installation, successful runs are described conservatively as following the loader's sequential `bf16`-then-4-bit attempt policy rather than as universally quantized executions.

Generation settings were shared across all conditions through `configs/prompts.yaml`: temperature was fixed at `0.0`, `do_sample` was set to `false`, batch size was `1`, and the global seed was `42`. Decoding was therefore deterministic rather than stochastic. Condition-specific maximum generation lengths were defined in `configs/datasets.yaml`: 512 new tokens for GSM8K, 1024 for MATH L1–L3, and 256 for both ARC-Challenge and TruthfulQA MC1.

Prompt construction and answer extraction were both handled through shared interfaces. Prompt generation was implemented in `prompts/builder.py`, which assembled the strategy-specific wrapper, inserted dataset-specific few-shot demonstrations when applicable, appended dataset-level answer-format rules, and then optionally applied the tokenizer chat template. GSM8K used the `gsm8k` extractor, MATH L1–L3 used the `math` extractor, ARC-Challenge used the `arc` extractor, and TruthfulQA MC1 used the `truthfulqa` extractor. The runner passed the generated response to `extract_answer(...)` and then scored the extracted prediction against the gold answer using `grade_prediction(...)`. Multiple-choice outputs were therefore mapped to final labels through task-specific extraction and grading rules, while free-form mathematical outputs were scored through the corresponding math-oriented normalization and extraction path defined in the evaluation code.

Latency and VRAM measurements in this study should be interpreted as environment-dependent deployment measurements rather than hardware-invariant properties of the models. They depend on factors such as GPU type, precision mode, batch size, quantization behavior, and software stack. Nevertheless, under a fixed shared environment they remain informative for relative comparison across model–prompt configurations, which is the role they play in the present deployment-aware analysis.

For transparency and reproducibility, we provide the benchmark code, prompt configuration, model configuration, and curated evaluation summaries at `https://anonymous.4open.science/r/UDAE-D371/`.

3.5 Evaluation metrics

We report the following metrics.

Accuracy is the fraction of correctly answered examples for a condition. **Wilson confidence interval** provides uncertainty bounds around the accuracy estimate. **Mean latency** is the average inference time in seconds for a condition. **Peak VRAM** is the maximum observed memory usage in gigabytes for the condition. **Mean tokens per second** is the observed output throughput.

In addition to per-condition results, we compute a weighted accuracy summary using the following task weights: GSM8K 0.40, MATH L1–L3 0.30, ARC-Challenge 0.20, and TruthfulQA MC1 0.10. This weighting is intended as a compact deployment-oriented summary that places somewhat greater emphasis on mathematical reasoning while still retaining science-oriented multiple-choice reasoning and truthfulness evaluation in the aggregate score. We do not treat this vector as uniquely correct, and we do not interpret the weighted score as a complete account of model quality.

To assess whether the headline conclusion is overly dependent on this choice, we also examined alternative weighting schemes, including equal weighting and task-heavy variants. In the sensitivity analysis, the exact mid-table order changes across schemes, but two broad conclusions remain stable: Gemma-4-26B-A4B with zero-shot prompting remains the weighted leader in the paper-original, equal-weight, GSM8K-heavy, MATH-heavy, and ARC-heavy schemes, whereas truthfulness-heavy weighting shifts the top position toward Qwen-family few-shot CoT configurations. At the same time, Gemma-4-E4B remains consistently near the top across weighting schemes and continues to offer a markedly stronger latency–memory tradeoff than the heaviest high-scoring alternatives. We therefore use the paper-original weighting as a descriptive summary rather than as a claim that one aggregate vector is universally preferred.

To strengthen the deployment-aware interpretation, we also report additional analyses built on the weighted summary. These include bootstrap confidence intervals and paired permutation comparisons for the strongest weighted configurations, Pareto-frontier analysis over weighted accuracy, latency, and VRAM, resource-normalized efficiency metrics, deployment-budget summaries, prompt-instability measures across strategies, and compatibility diagnostics for cases where the shared prompt-and-extraction pipeline may interact unevenly with specific model families. These analyses are used to distinguish score leadership from practical operating-point quality and to assess whether observed ranking differences remain meaningful under resource constraints and prompt variation.

4 Results

This section reports the empirical findings of the unified 238-example evaluation. We begin with an overview of the complete matched-size result set and the weighted ranking across model–prompt configurations. We then move from leaderboard-style summaries to deployment-aware interpretation by examining benchmark-specific behavior, Pareto-efficient operating points, routing headroom, and compatibility diagnostics. The goal is not only to identify which configuration attains the highest weighted score, but also to determine which configurations remain attractive once latency, memory, prompting sensitivity, and task-specific behavior are considered together.

4.1 Overview of the unified result set

The final result set contains all 84 model–dataset–strategy conditions under the same 238-example protocol. This matters because the analysis no longer mixes different sample sizes across tasks or models. Every weighted comparison and every per-dataset comparison is therefore based on the same number of examples per condition.

Table 4 provides a compact summary of the most relevant weighted and benchmark-specific results, while the complete 84-condition matrix is reported in Appendix C. Table 1 presents the top weighted configurations across the 21 model–strategy combinations, and Figure 2 visualizes the same ranking. Figures 3a, 3b, and 3c provide the main views of dataset-specific performance across prompting strategies. The later results

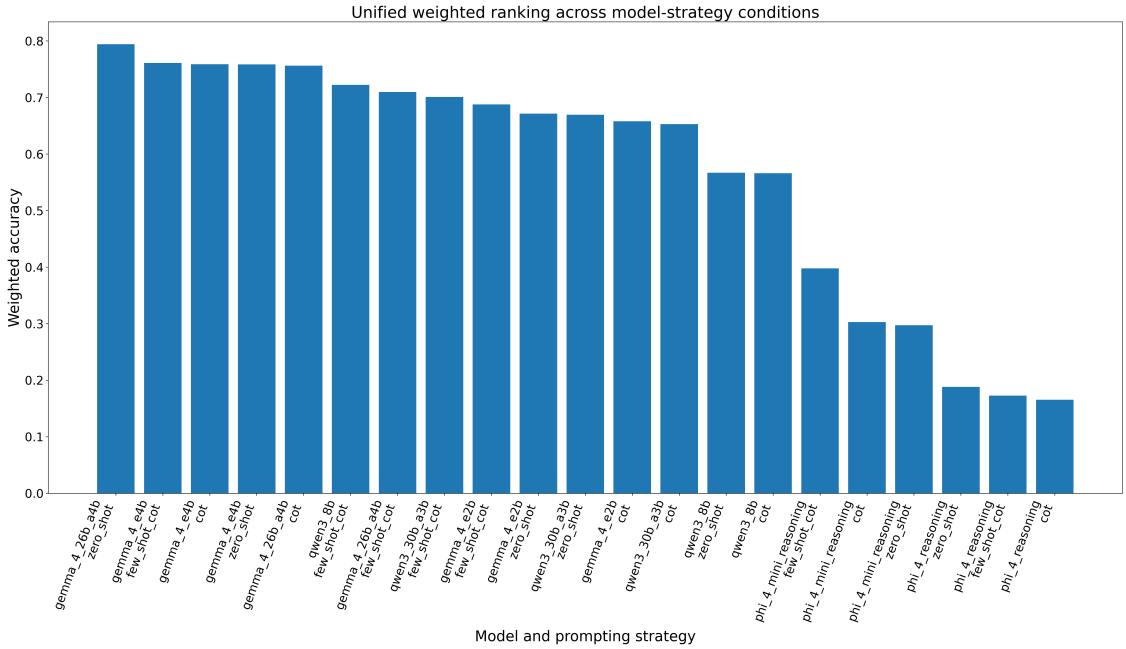


Figure 2: Weighted ranking across model–strategy configurations under the unified 238-example protocol. Gemma-4-26B-A4B with zero-shot prompting attains the highest weighted score, while Gemma-4-E4B remains near the top across all three prompting settings, suggesting that the highest-score configuration and the most deployment-attractive operating point may differ.

subsections then extend this overview in three directions: deployment-aware operating points, oracle routing headroom, and compatibility diagnostics. In particular, Section 4.4 formalizes latency–memory–accuracy tradeoffs, Section 4.5 quantifies task-aware routing headroom, and Section 4.6 examines failure patterns that are not visible from accuracy alone.

Table 1 already shows the central tension of the paper. Gemma-4-26B-A4B with zero-shot prompting achieves the highest weighted score at 0.794, but Gemma-4-E4B remains very close to the top across all three prompting strategies while requiring substantially less latency and memory. This means that the score leader and the strongest practical operating point are not obviously the same configuration, even before the more formal deployment-aware analyses are introduced.

Table 1: Top weighted configurations under the unified 238-example protocol. Values are taken from the weighted summary produced from the complete $7 \times 4 \times 3$ evaluation matrix. The table highlights that the highest weighted-score configuration and the strongest practical operating point are not necessarily the same.

Model	**Strategy**	**Weighted Acc.**	**Latency (s)**	**VRAM (GB)**	**TPS**
Gemma-4-26B-A4B	Zero-shot	0.794	7.283	48.067	20.570
Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895	25.850
Gemma-4-E4B	CoT	0.759	4.720	14.895	25.615
Gemma-4-E4B	Zero-shot	0.758	4.373	14.895	25.684
Gemma-4-26B-A4B	CoT	0.756	8.715	48.067	19.960
Qwen3-8B	Few-shot CoT	0.722	7.025	15.256	49.419
Gemma-4-26B-A4B	Few-shot CoT	0.710	10.580	48.067	19.598
Qwen3-30B-A3B	Few-shot CoT	0.701	14.701	57.621	25.127
Gemma-4-E2B	Few-shot CoT	0.688	5.997	9.543	31.747
Gemma-4-E2B	Zero-shot	0.671	5.098	9.543	31.648

4.2 Weighted ranking under the unified protocol

The weighted ranking is reported in Table 1 and shown in Figure 2. The highest weighted score is achieved by **Gemma-4-26B-A4B with zero-shot prompting** at **0.794**. The next strongest configurations are all variants of **Gemma-4-E4B**: few-shot CoT at **0.761**, CoT at **0.759**, and zero-shot at **0.758**. **Gemma-4-26B-A4B with CoT** follows at **0.756**. The strongest Qwen condition is **Qwen3-8B with few-shot CoT** at **0.722**, followed by **Qwen3-30B-A3B with few-shot CoT** at **0.701**.

The weighted ranking identifies a score leader, but it also shows that the strongest Gemma-4-E4B operating points remain close to that lead. The difference between the top weighted configuration and Gemma-4-E4B few-shot CoT is **0.033**. Bootstrap intervals for the strongest weighted configurations overlap, which cautions against treating that gap as overwhelmingly large in practical terms. However, interval overlap is only a rough descriptive heuristic, whereas the paired permutation analysis evaluates aligned example-level differences directly. Under that stronger paired test, Gemma-4-26B-A4B zero-shot remains significantly ahead of Gemma-4-E4B few-shot CoT, CoT, and zero-shot under the current weighted metric, with two-sided permutation p-values of approximately **0.024**, **0.016**, and **0.015**, respectively. The combined interpretation is therefore more nuanced than a simple leaderboard reading: the weighted leader is real, but the margin is still modest enough that deployment cost remains central to model selection.

At the bottom of the weighted table, the Phi family is clearly weaker overall. Phi-4-Mini-Reasoning peaks at **0.398** under few-shot CoT. Phi-4-Reasoning remains lower still, with weighted scores of **0.189**, **0.173**, and **0.166** across zero-shot, few-shot CoT, and CoT, respectively. The weighted ranking therefore establishes a score leader, but it does not by itself determine the most deployment-attractive configuration. That question is examined next through deployment-aware operating-point analysis.

4.3 Dataset-specific performance

The unified full matrix in Table 5 and the benchmark-level summary in Table 2 show large differences across datasets. The heatmaps in Figure 3 provide the corresponding model–dataset–strategy view across the three prompting conditions.

On **ARC-Challenge**, the strongest condition is Gemma-4-26B-A4B zero-shot with accuracy **0.945**. Gemma-4-26B-A4B few-shot CoT follows at **0.937**. Gemma-4-E4B zero-shot reaches **0.899**, and both CoT and few-shot CoT are at **0.891**. Qwen and Phi models are much lower on this dataset. Under the present prompt family, ARC-Challenge is therefore especially favorable to the strongest Gemma configurations.

On **GSM8K**, the strongest condition is Qwen3-8B few-shot CoT at **0.819**. Qwen3-30B-A3B few-shot CoT follows at **0.807**. Gemma-4-26B-A4B zero-shot reaches **0.794**, Gemma-4-E4B CoT and zero-shot both reach **0.790**, and Gemma-4-26B-A4B CoT reaches **0.782**. Here the ordering differs markedly from ARC-Challenge: Qwen becomes much stronger relative to Gemma on this benchmark.

The Phi-4-Reasoning results on GSM8K require special caution. Their accuracies are strikingly low under all three prompting strategies, and the compatibility diagnostics in Section 4.6 show that this is best understood as a robustness failure under the unified interface rather than as a clean standalone estimate of mathematical reasoning quality. In particular, Phi-4-Reasoning exhibits extremely high missing-prediction rates on GSM8K, with **NaN prediction rates between 0.958 and 0.983**, together with near-universal `<think>` traces under the present shared pipeline. Whatever internal reasoning capacity the model may possess, its inability to reliably produce scoreable final outputs under the shared protocol makes it unsuitable for out-of-the-box deployment on this task family in the present setting.

On **MATH L1–L3**, the strongest condition is Gemma-4-E4B few-shot CoT at **0.693**. Gemma-4-26B-A4B zero-shot reaches the same value to three decimal places in the aggregated output, while Gemma-4-E2B few-shot CoT reaches **0.689**. Qwen3-8B few-shot CoT reaches **0.639**, and Qwen3-30B-A3B few-shot CoT reaches **0.613**. This dataset produces a tighter cluster among the stronger Gemma configurations, with a weaker but still competitive Qwen group.

On **TruthfulQA MC1**, performance is near ceiling for several models. Phi-4-Reasoning few-shot CoT reaches **1.000**. Phi-4-Reasoning CoT and zero-shot both reach **0.996**. Qwen3-30B-A3B CoT reaches **0.987**,

and Qwen3-8B CoT also reaches **0.987**. Phi-4-Mini-Reasoning CoT reaches **0.983**. Gemma models are lower on this dataset, with Gemma-4-26B-A4B between **0.727** and **0.794** depending on prompting and Gemma-4-E4B between **0.643** and **0.739**.

These results show that no single model family dominates all task types. The strongest model on one dataset is not necessarily strongest on another. That benchmark-specific complementarity is one reason why a single aggregate score is not sufficient for deployment-aware interpretation, and it also motivates the oracle routing upper-bound analysis reported later in Section 4.5.

Table 2: Best-performing condition for each dataset under the unified 238-example protocol.

Dataset	**Model**	**Strategy**	**Accuracy**	n
ARC-Challenge	Gemma-4-26B-A4B	Zero-shot	0.945	238
GSM8K	Qwen3-8B	Few-shot CoT	0.819	238
MATH L1–L3	Gemma-4-E4B	Few-shot CoT	0.693	238
TruthfulQA MC1	Phi-4-Reasoning	Few-shot CoT	1.000	238

4.4 Deployment-aware operating points

A single weighted winner does not settle the deployment question. Table 3 and Figures 4 and 5 show that the weighted leader and the strongest practical operating point are not the same. The weighted winner, Gemma-4-26B-A4B zero-shot, uses **48.067 GB** of VRAM and has mean latency **7.283 s** across tasks. By contrast, Gemma-4-E4B few-shot CoT reaches a weighted score of **0.761** with mean latency **3.677 s** and mean VRAM **14.895 GB**. Gemma-4-E4B CoT and zero-shot show similar tradeoffs. In practical terms, Gemma-4-E4B gives up only a modest amount of weighted score while reducing both latency and memory substantially.

This contrast becomes clearer under Pareto-frontier analysis. The Pareto-efficient set includes **Gemma-4-26B-A4B zero-shot**, **Gemma-4-E4B few-shot CoT**, and several **Gemma-4-E2B** variants, along with much lower-scoring but lighter Phi-4-Mini-Reasoning conditions. No single configuration therefore dominates the joint accuracy–latency–memory space. Gemma-4-26B-A4B zero-shot remains the heavy high-score operating point, whereas Gemma-4-E4B few-shot CoT is the strongest *high-performing practical* operating point under the present deployment-aware view.

Resource-normalized metrics provide a different perspective. By the combined efficiency score based on weighted accuracy normalized by latency and memory, the strongest condition is **Gemma-4-E2B CoT**, followed by **Gemma-4-E4B few-shot CoT** and other Gemma-4-E2B variants. This does not make Gemma-4-E2B the best overall model. Instead, it shows that score leadership, practical high-performance, and efficiency leadership are distinct criteria that can favor different configurations.

The deployment-budget summary sharpens this point further. Under **16 GB**, **24 GB**, and **48 GB** memory budgets, the strongest weighted configuration is consistently **Gemma-4-E4B few-shot CoT**. Only in the unrestricted setting does **Gemma-4-26B-A4B zero-shot** recover the top position. Thus, the answer to "which model is best" depends directly on the deployment budget.

The Qwen family reveals a different tradeoff. Qwen3-8B few-shot CoT reaches **0.722** weighted accuracy with mean latency **7.025 s** and mean VRAM **15.256 GB**. Qwen3-30B-A3B few-shot CoT reaches **0.701** with much higher memory use at **57.621 GB** and much higher latency at **14.701 s**. Under the present weighted metric, the 8B Qwen configuration is therefore much more attractive than the larger A3B configuration.

Taken together, these results support the paper's central deployment-aware message: the highest weighted-score configuration is not automatically the most attractive operating point once latency, memory, efficiency, and budget constraints are considered directly.

For reference, the corresponding budget and efficiency summaries are reported in Appendix D, Table 9.

Table 3: Deployment-oriented comparison among top weighted configurations. The weighted leader and the strongest practical operating point are not the same.

Model	Strategy	Weighted Acc.	Latency (s)	VRAM (GB)
Gemma-4-26B-A4B	Zero-shot	0.794	7.283	48.067
Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895
Gemma-4-E4B	CoT	0.759	4.720	14.895
Gemma-4-E4B	Zero-shot	0.758	4.373	14.895
Qwen3-8B	Few-shot CoT	0.722	7.025	15.256

4.5 Oracle routing upper bound

The benchmark-specific winners in Table 2 suggest measurable cross-task complementarity. That observation motivates a simple oracle routing upper-bound analysis. For each benchmark, we select the best-performing configuration and then compute the resulting weighted aggregate score under the same paper-original weighting rule. This oracle task-aware selector achieves a weighted score of **0.825**, compared with **0.794** for the best single fixed configuration, Gemma-4-26B-A4B zero-shot. The gain of approximately **0.031** shows that the benchmark complementarity is not merely qualitative; it is large enough to produce measurable aggregate headroom.

This result should be interpreted carefully. The oracle selector is not a deployed routing system, and it does not include the overhead of classifier uncertainty, routing latency, or implementation complexity. Its purpose is instead to quantify an upper bound on what task-aware selection could achieve under the observed benchmark complementarity. In that sense, it supports a future direction for deployment-aware routing without changing the main identity of the present paper into an ensemble-learning study.

The oracle result is also suggestive from a systems perspective. Because the per-dataset winners are not all drawn from the heaviest model, the observed complementarity indicates that selective routing could potentially improve aggregate performance without requiring every query to be handled by the most resource-intensive configuration. This strengthens the practical argument for lightweight task-aware or budget-aware routing as a future direction, while keeping the present analysis clearly in the upper-bound category.

4.6 Compatibility and failure diagnostics

The Phi-4-Reasoning results merit dedicated diagnostic analysis because the model performs near ceiling on TruthfulQA MC1 yet fails badly on GSM8K and MATH L1–L3 under the shared pipeline. The compatibility statistics show that these failures are highly task-dependent and are strongly associated with interface-adherence and extraction problems under the shared pipeline. On **GSM8K**, Phi-4-Reasoning exhibits **NaN prediction rates of 0.958 to 0.983** across prompting strategies, together with think-tag prevalence near 1.0 and mean response lengths of roughly **1,936** to **2,002** characters. On **MATH L1–L3**, think-tag prevalence remains above **0.92** for all three prompting strategies, mean response length rises to roughly **3,505** to **3,683** characters, and malformed-output rates remain around **0.10** to **0.12**. By contrast, on **TruthfulQA MC1**, the model achieves near-perfect accuracy with no missing-prediction problem.

The error analysis reinforces the view that this is a deployment-relevant interface robustness problem under the unified protocol. For Phi-4-Reasoning on GSM8K, the dominant error type is **extraction failure**, with counts of **234**, **232**, and **228** under CoT, few-shot CoT, and zero-shot, respectively. On MATH L1–L3 and ARC-Challenge, the dominant failure category is **pipeline compatibility or think-trace issue**, followed by smaller but still visible counts of **final-answer format failure**. Representative sampled outputs show repeated instruction echoing, long internal-style traces, and outputs that do not cleanly terminate in the required extractable final-answer format. This means that, under the present shared interface, Phi-4-Reasoning is not robust enough for reliable out-of-the-box deployment on these tasks, regardless of whether the underlying problem is framed as extraction mismatch, formatting instability, or broader prompt-and-pipeline incompatibility.

These diagnostics matter for two reasons. First, they explain why Phi-4-Reasoning can appear exceptionally weak on some benchmarks while remaining extremely strong on TruthfulQA MC1. Second, they illustrate a broader methodological point of the paper: under a unified benchmark pipeline, some model families can interact unevenly with shared prompting and extraction rules. That interaction is itself part of deployment-aware evaluation and should not be hidden behind a single aggregate score.

A compact tabular summary of these compatibility statistics is reported in Appendix D, Table 10.

4.7 Prompt sensitivity

Figure 6 summarizes how weighted performance changes across prompting strategies, and the rank-instability analysis quantifies the same pattern more formally. The overall weighted ranking is comparatively stable between CoT and zero-shot prompting, with Spearman $\rho = \mathbf{0.964}$ and Kendall $\tau = \mathbf{0.905}$. However, the agreement weakens meaningfully when few-shot CoT is involved: CoT versus few-shot CoT yields Spearman $\rho = \mathbf{0.750}$ and Kendall $\tau = \mathbf{0.619}$, while few-shot CoT versus zero-shot yields Spearman $\rho = \mathbf{0.679}$ and Kendall $\tau = \mathbf{0.524}$. Prompting therefore changes the ranking structure rather than merely shifting all models in the same direction.

The model-level view makes the same point. Gemma-4-E4B is relatively stable across prompting strategies, with weighted scores tightly clustered near the top. By contrast, **Qwen3-8B** is strongly prompt-sensitive: its weighted rank shifts from **5** under CoT to **2** under few-shot CoT and back to **5** under zero-shot, giving it the largest weighted rank range in the study. Gemma-4-26B-A4B also changes rank across prompts, although more moderately, moving from rank **2** under CoT to rank **1** under zero-shot. Gemma-4-E2B changes from rank **3** under CoT and zero-shot to rank **5** under few-shot CoT. By contrast, Qwen3-30B-A3B and both Phi families remain comparatively fixed in weighted rank, although that stability is not always desirable because it can reflect consistently weak performance rather than robustness.

Dataset-specific rank correlations show a similar pattern. ARC-Challenge and TruthfulQA MC1 are comparatively stable across prompting strategies, whereas GSM8K and MATH L1–L3 show larger rank-order changes, especially when few-shot CoT is compared with zero-shot prompting. This matters because it shows that prompting strategy is not a minor implementation choice. Under a unified evaluation protocol, it is an experimental factor that can materially alter the relative ordering of model families.

4.8 Latency and memory trends

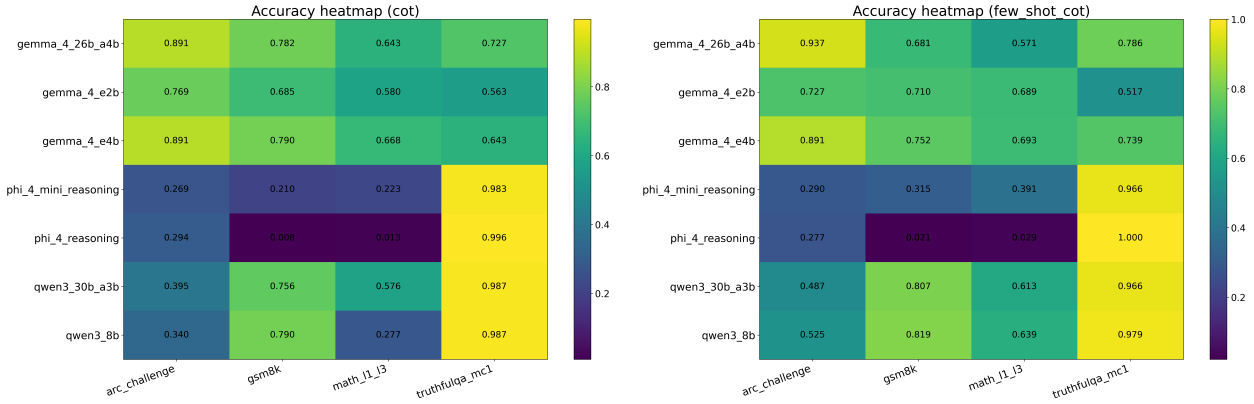
Deployment cost differs sharply across models, and Figures 4 and 5 visualize those differences directly. The lowest mean VRAM among the evaluated models belongs to Phi-4-Mini-Reasoning at **7.145 GB**, followed by Gemma-4-E2B at **9.543 GB**, Gemma-4-E4B at **14.895 GB**, Qwen3-8B at **15.256 GB**, Phi-4-Reasoning at **27.305 GB**, Gemma-4-26B-A4B at **48.067 GB**, and Qwen3-30B-A3B at **57.621 GB**.

These memory differences are paired with substantial latency differences. Gemma-4-E4B operates in the range of roughly **3.7** to **4.7 s** mean latency across tasks depending on prompting. Gemma-4-E2B operates around **4.4** to **6.0 s**. Qwen3-8B operates around **7.0** to **7.7 s**. Gemma-4-26B-A4B operates around **7.3** to **10.6 s**. Qwen3-30B-A3B operates around **14.7** to **15.3 s**. Phi-4-Reasoning remains around **9.6 s** despite much weaker weighted performance.

These trends reinforce the broader argument of the results section. Accuracy and weighted accuracy remain necessary, but they are not sufficient for deployment-aware interpretation. Once latency and memory are considered jointly, the result set is better understood as a collection of operating points with different tradeoffs rather than as a single global ranking.

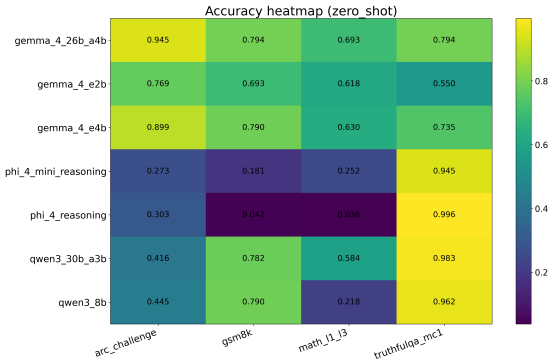
5 Discussion

The central result of this study is not simply that one model configuration ranks first. The more important finding is that a fully unified protocol changes how open-model benchmarks should be interpreted. Under the matched-size design, Gemma-4-26B-A4B with zero-shot prompting is the weighted score leader, but



(a) CoT

(b) Few-shot CoT



(c) Zero-shot

Figure 3: Accuracy heatmaps across datasets, models, and prompting strategies. These plots show that task-specific patterns remain large across model families and that prompting changes relative ordering.

Gemma-4-E4B remains close to the top across all three prompting settings while requiring substantially less latency and memory. The deployment-budget analysis further shows that Gemma-4-E4B few-shot CoT is the strongest weighted configuration under 16 GB, 24 GB, and 48 GB memory constraints. The main lesson is therefore that open-model evaluation should distinguish between score leadership and deployment-attractive operating points.

The statistical analysis supports that interpretation. Bootstrap confidence intervals for the strongest weighted configurations overlap, which cautions against reading the top rows as separated by overwhelming practical margins. At the same time, paired permutation tests show that Gemma-4-26B-A4B zero-shot remains significantly ahead of the strongest Gemma-4-E4B variants under the current weighted metric. The appropriate conclusion is therefore not that the top configurations are equivalent, but that the score leader's advantage is modest relative to the corresponding resource differences. This is precisely the kind of result for which operating-point analysis is more informative than a single aggregate ranking.

The deployment-aware analyses make this point more formal. The Pareto frontier shows that no single configuration dominates the full accuracy–latency–memory space. Gemma-4-26B-A4B zero-shot occupies the heavy high-score end of that frontier, Gemma-4-E4B few-shot CoT is the strongest practical high-performing point, and several Gemma-4-E2B variants emerge as attractive efficiency-oriented alternatives. In other words, score leadership, practical high performance, and efficiency leadership are not the same objective, and they need not identify the same model.

Prompting and benchmark composition also matter. Prompting is not a minor implementation choice: under the unified protocol, it changes rank order, especially when few-shot CoT is introduced. Benchmark-specific behavior remains substantial as well. ARC-Challenge favors the strongest Gemma configurations, GSM8K

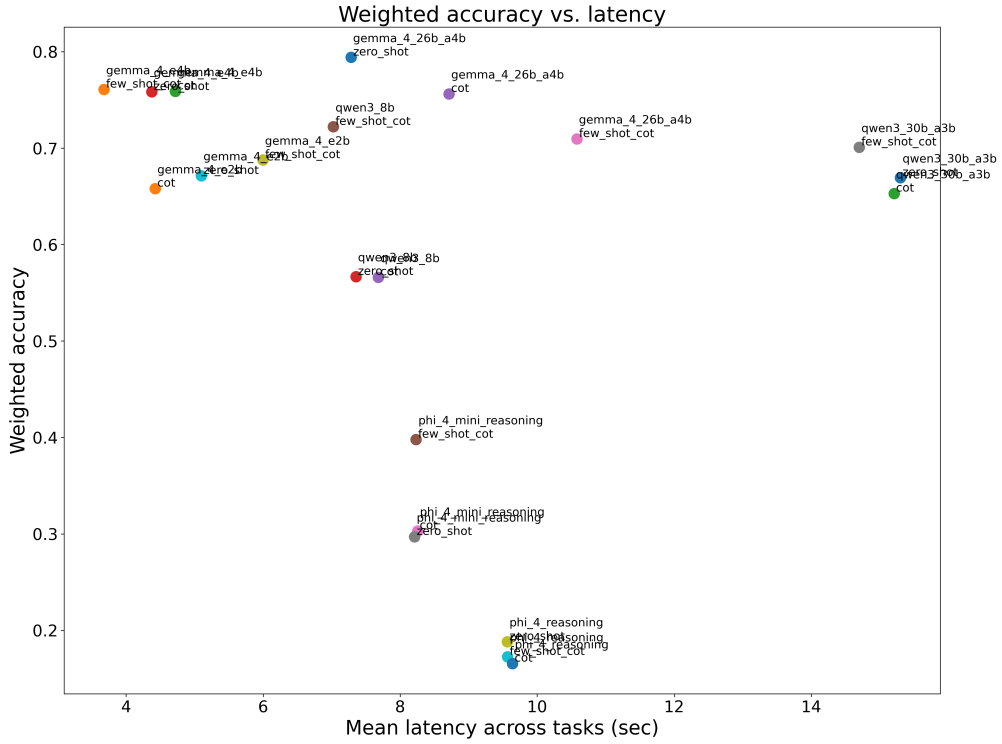


Figure 4: Weighted accuracy versus mean latency across model-strategy configurations. Gemma-4-E4B lies close to the top of the weighted ranking while remaining much faster than Gemma-4-26B-A4B and Qwen3-30B-A3B.

favors Qwen more strongly, MATH L1–L3 yields a tighter competition among the top Gemma variants, and TruthfulQA MC1 places several models near ceiling. This complementarity is one reason why a single aggregate score can be misleading in isolation, and it also explains the oracle routing upper bound. The oracle selector raises the weighted score from 0.794 to 0.825, showing measurable headroom for future lightweight task-aware or budget-aware routing without making routing the main contribution of the present paper.

The Phi-4-Reasoning results provide a different methodological lesson. If one looked only at TruthfulQA MC1, Phi-4-Reasoning would appear exceptionally strong. Under the unified benchmark as a whole, however, its overall weighted performance is poor because GSM8K and MATH L1–L3 expose severe robustness failures under the shared interface. The diagnostic analysis shows that these failures are dominated by extraction and formatting problems, long think-style traces, and broader prompt-and-pipeline mismatch. From a deployment perspective, that distinction matters less than the practical outcome: a model that cannot reliably produce scoreable outputs under a simple unified protocol is not robust enough for out-of-the-box use on those tasks. This is precisely why unified evaluation should consider not only accuracy, but also interface adherence and failure behavior.

Overall, the paper argues for a shift in evaluation style. Open reasoning model benchmarking is most informative when it is unified, balanced, deployment-aware, and explicit about prompt sensitivity, resource constraints, and failure modes. Under that view, the practical question is not only "which model scores highest," but also "which model remains attractive under realistic memory, latency, and interface constraints."

Limitations

This study has several limitations. The benchmark suite includes four tasks, which is enough to expose meaningful variation but not enough to represent the full range of modern LLM workloads. The unified sample size is 238 examples per condition because the current shared preparation pipeline supports a largest

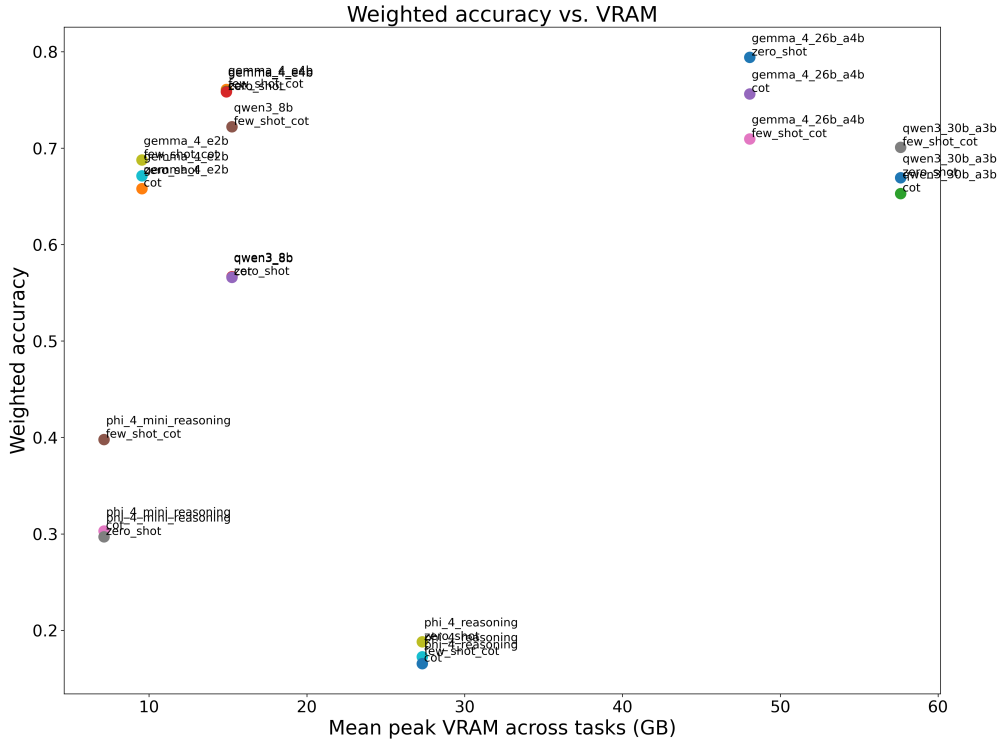


Figure 5: Weighted accuracy versus mean VRAM across model-strategy configurations. The figure highlights that the weighted leader is not the most resource-efficient operating point.

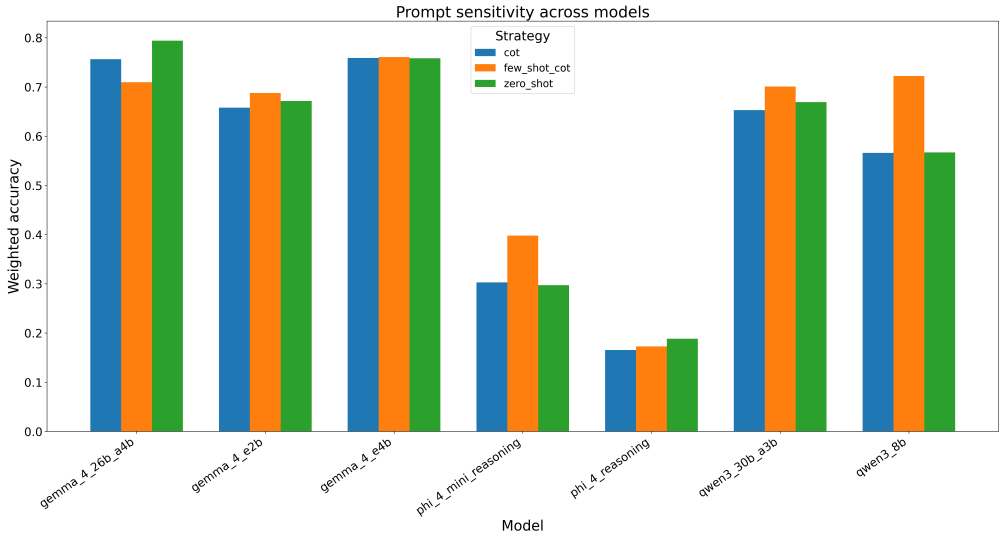


Figure 6: Prompt sensitivity across models. Prompting strategy changes rank order rather than providing a uniform gain for all model families.

common matched subset of 238 examples across all four benchmarks, with `MATH L1-L3` determining that cap after filtering to levels 1–3. This improves cross-condition comparability, but it also means that the present study prioritizes protocol consistency over full-split coverage and does not exploit the full size of every benchmark split. The weighted aggregate likewise depends on a chosen task-weight vector. That vector is useful for compact summary, but it is not uniquely correct, and alternative weighting schemes do

Table 4: Compact summary of the most relevant results under the unified 238-example protocol. The top block reports weighted aggregate operating points across datasets, while the bottom block reports the best-performing condition for each individual benchmark. The complete 84-condition matrix is provided in Appendix C.

Role / Summary	Dataset	Model	Strategy	Accuracy	95% CI	Latency (s)	VRAM (GB)
Weighted-score leader	Overall	Gemma-4-26B-A4B	Zero-shot	0.794	–	7.283	48.067
Practical operating point	Overall	Gemma-4-E4B	Few-shot CoT	0.761†	–	3.677	14.895
Strongest Qwen condition	Overall	Qwen3-8B	Few-shot CoT	0.722†	–	7.025	15.256
Strongest Phi condition	Overall	Phi-4-Mini-Reasoning	Few-shot CoT	0.398†	–	8.232	7.145
Best per dataset	ARC-Challenge	Gemma-4-26B-A4B	Zero-shot	0.945	[0.909, 0.968]	3.433	48.067
Best per dataset	GSM8K	Qwen3-8B	Few-shot CoT	0.819	[0.765, 0.863]	6.674	15.256
Best per dataset	MATH L1–L3	Gemma-4-E4B	Few-shot CoT	0.693	[0.632, 0.748]	8.278	14.895
Best per dataset	TruthfulQA MC1	Phi-4-Reasoning	Few-shot CoT	1.000	[0.984, 1.000]	4.362	27.305

† Weighted accuracy from the unified aggregate summary rather than single-dataset accuracy. Confidence intervals are not shown for weighted scores because the weighted aggregate combines results across datasets.

change parts of the ranking table, especially when truthfulness is emphasized more heavily. The study also uses a single evaluation stack and a single hardware environment, so absolute latency and memory values should be interpreted as environment-dependent measurements rather than universally fixed deployment costs. Although the relative trends are still informative under a shared environment, the exact operating points may shift under different hardware, precision modes, batch sizes, or quantization behavior. Finally, the oracle routing result is only an upper bound. It demonstrates routing headroom, but it does not model the overhead, uncertainty, or engineering complexity of an actual deployed selector.

Future Work

Future work should extend the benchmark to additional task families such as broader knowledge evaluation, instruction following, code generation, and robustness under format constraints. It should also study cross-hardware behavior and cost-constrained operating envelopes more directly. A particularly promising direction is lightweight task-aware or budget-aware routing, in which an efficient selector chooses among a small set of strong operating points rather than relying on heavy ensemble inference. Another priority is to improve model-family-aware answer extraction and compatibility diagnostics, especially for reasoning models that emit internal-style traces or deviate from the expected final-answer format under a shared benchmark interface. More broadly, future work should continue to develop evaluation methods that treat open-model selection as a multi-objective deployment problem rather than as a leaderboard exercise alone.

6 Conclusion

This paper presented a unified deployment-aware evaluation of seven open reasoning language model configurations across four benchmark tasks and three prompting strategies, with all 84 conditions tested under the same 238-example protocol. Under this matched-size design, Gemma-4-26B-A4B with zero-shot prompting achieved the highest weighted score, but Gemma-4-E4B emerged as the strongest practical high-performing operating point once latency, memory, efficiency, and deployment-budget constraints were taken into ac-

count. The study also showed that prompting strategy changes ranking order, that benchmark-specific complementarity creates measurable routing headroom, and that some apparent failures, especially for Phi-4-Reasoning, reflect deployment-relevant interface robustness problems under the shared evaluation pipeline. Taken together, these results support a single conclusion: for open reasoning language models, evaluation should move beyond single-score leaderboard ranking toward deployment-aware operating-point analysis.

References

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Besmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, Vibhav Vineet, Yue Wu, Safoora Yousefi, and Guoqing Zheng. Phi-4-reasoning technical report. Microsoft Research, 2025. URL `https://www.microsoft.com/en-us/research/publication/phi-4-reasoning-technical-report/`.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. *ACM Computing Surveys*, 56(9), 2024.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

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. *arXiv preprint arXiv:2110.14168*, 2021.

Chengguang Gan and Tatsunori Mori. Sensitivity and robustness of large language models to prompt template in japanese text classification tasks. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pp. 1–11, Hong Kong, China, 2023.

Google. Gemma 4 model overview. Google AI for Developers, 2026a. URL `https://ai.google.dev/gemma/docs/core`.

Google. Gemma 4: Our most capable open models to date. Google Blog, 2026b. URL `https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/`.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22199–22213, 2022.

Md Tahmid Rahman Laskar, Sawsan Alqahtani, M. Saiful Bari, Mizanur Rahman, Mohammad Abdullah Matin Khan, Haidar Khan, Israt Jahan, Amran Bhuiyan, Chee Wei Tan, Md Rizwan Parvez, Enamul Hoque, Shafiq Joty, and Jimmy Huang. A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

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 Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, 2022.

Microsoft. Phi-4-mini-reasoning model card. Hugging Face, 2025. URL https://huggingface.co/microsoft/Phi-4-mini-reasoning.

Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*, 2024.

Qwen Team. Qwen3-30b-a3b model card. Hugging Face, 2025a. URL https://huggingface.co/Qwen/Qwen3-30B-A3B.

Qwen Team. Qwen3-8b model card. Hugging Face, 2025b. URL https://huggingface.co/Qwen/Qwen3-8B.

Qwen Team. Qwen3: Think deeper, act faster. Qwen Blog, 2025c. URL https://qwenlm.github.io/blog/qwen3/.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*, 2024.

Tianyi Tang, Yiwen Hu, Bingqian Li, Wenyang Luo, Zijing Qin, Haoxiang Sun, Jiapeng Wang, Shiyi Xu, Xiaoxue Cheng, Geyang Guo, Han Peng, Bowen Zheng, Yiru Tang, Yingqian Min, Yushuo Chen, Jie Chen, Yuanqian Zhao, Luran Ding, Yuhao Wang, Zican Dong, Chunxuan Xia, Junyi Li, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. Llmbox: A comprehensive library for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (System Demonstrations)*, 2024.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Zhenqiang Gong, and Xing Xie. Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*, 2023.

Appendix

A Prompt Templates

This appendix reports the exact prompt construction procedure used in the unified evaluation pipeline. Prompt wording was held fixed within each prompting strategy so that all model–dataset–strategy conditions were evaluated under the same prompt family. The wrapper templates were defined in `configs/prompts.yaml`, while the few-shot demonstration examples were loaded from `prompts/few_shot_examples.json` through the shared prompt builder in `prompts/builder.py`.

A.1 Zero-shot prompt template

The zero-shot template was:

```
{question}
Answer:
```

A.2 Chain-of-thought (CoT) prompt template

The CoT template was:

```
{question}
Let's think step by step.
```

A.3 Few-shot chain-of-thought (few-shot CoT) prompt template

The few-shot CoT wrapper template was:

```
{few_shot_block}
Q: {question}
A: Let's think step by step.
```

The evaluation code constructed `{few_shot_block}` by loading dataset-specific demonstrations from `prompts/few_shot_examples.json`. For each example, the block format was:

```
Q: {example question}
A: {example reasoning}
#### {example answer}
```

The resulting few-shot demonstration blocks used in the experiments were as follows.

A.3.1 GSM8K few-shot demonstration block

```
Q: A store sold 12 notebooks on Monday and 15 notebooks on Tuesday.  How
many notebooks were sold in total?
A: We add the notebooks sold on both days:  12 + 15 = 27.
#### 27

Q: Sara has 18 apples and gives 7 away.  How many apples does she have
left?
A: Subtract the apples she gave away:  18 - 7 = 11.
#### 11

Q: A box contains 4 rows of pencils with 6 pencils in each row.  How many
pencils are there?
A: Multiply rows by pencils per row:  4 x 6 = 24.
#### 24
```

A.3.2 MATH L1–L3 few-shot demonstration block

```
Q: Compute 7 + 8.
A: Adding 7 and 8 gives 15.
#### 15

Q: What is 3^2?
A: 3 squared means 3 multiplied by itself:  3 x 3 = 9.
#### 9

Q: If x + 2 = 5, find x.
A: Subtract 2 from both sides:  x = 5 - 2 = 3.
#### 3
```

A.3.3 ARC-Challenge few-shot demonstration block

```
Q: Which planet is known as the Red Planet?
A. Earth
B. Mars
C. Venus
D. Jupiter
A: Mars is commonly called the Red Planet because of its reddish
appearance.
#### B

Q: What gas do plants mostly use in photosynthesis?
A. Oxygen
B. Nitrogen
C. Carbon dioxide
D. Hydrogen
A: Plants use carbon dioxide during photosynthesis.
#### C

Q: Which part of the cell contains genetic material?
A. Nucleus
B. Membrane
C. Ribosome
D. Cytoplasm
A: The nucleus stores most of the cell's genetic material.
#### A
```

A.3.4 TruthfulQA MC1 few-shot demonstration block

```
Q: Which answer is more truthful?
A. The Earth revolves around the Sun.
B. The Sun revolves around the Earth.
A: Modern astronomy shows that the Earth revolves around the Sun.
#### A

Q: Which answer is more truthful?
A. Vaccines can help prevent some infectious diseases.
B. Vaccines never work.
A: Vaccines are widely used to prevent infectious diseases.
#### A

Q: Which answer is more truthful?
A. Humans need water to survive.
B. Humans can live forever without water.
A: Humans require water for survival.
#### A
```

A.4 Dataset-specific answer-format rules

After the strategy template was constructed, the prompt builder appended dataset-specific answer-format rules.

For GSM8K and MATH L1–L3, the appended rules were:

```
Rules:
1.  Reason in no more than 4 short steps.
2.  The final line must be exactly:  #### <answer>
3.  Do not output anything after that final line.
```

For ARC-Challenge and TruthfulQA MC1, the appended rules were:

```
Rules:
1. Return only one capital letter.
2. Valid answers are A, B, C, D, or E.
3. Do not output any explanation.
```

A.5 Chat-template wrapping

After prompt construction, the pipeline optionally wrapped the prompt using the tokenizer chat template by passing the full prompt as a single user message and enabling `add_generation_prompt=True`. This behavior was controlled by the `use_chat_template` field in the model configuration.

B Reproducibility Summary

The current project includes raw runs, aggregated condition tables, weighted summaries, prompt configurations, model configurations, and manuscript-supporting statistical summaries. The core files most directly relevant to the paper include the aggregated unified-condition results, the weighted summary, the prompt-sensitivity summaries, the deployment-oriented comparisons, and the supplemental statistical tables reported in Appendix D.

The benchmark code, configuration files, and curated result tables used in this study are available at `https://anonymous.4open.science/r/UDAE-D371/`. Manuscript-specific figures and typeset tables were prepared separately for paper writing and are maintained with the paper source rather than the public repository.

C Full Per-Condition Evaluation Matrix

This appendix reports the complete per-condition results under the unified 238-example protocol. The table includes all 84 model–dataset–strategy conditions and reports the sample size, accuracy, Wilson confidence interval bounds, mean latency, and peak VRAM. This full matrix complements the compact summary shown in Table 4 in the main text.

Table 5: Full per-condition results under the unified 238-example protocol.

Dataset	Model	Strategy	N	Accuracy	CI Low	CI High	Latency (s)	VRAM (GB)
ARC-Challenge	Gemma-4-26B-A4B	CoT	238	0.891	0.845	0.924	3.132	48.067
ARC-Challenge	Gemma-4-E2B	CoT	238	0.769	0.711	0.818	0.450	9.543
ARC-Challenge	Gemma-4-E4B	CoT	238	0.891	0.845	0.924	0.496	14.895
ARC-Challenge	Phi-4-Mini-Reasoning	CoT	238	0.269	0.217	0.329	4.119	7.145
ARC-Challenge	Phi-4-Reasoning	CoT	238	0.294	0.240	0.355	4.830	27.305
ARC-Challenge	Qwen3-30B-A3B	CoT	238	0.395	0.335	0.458	10.307	57.621
ARC-Challenge	Qwen3-8B	CoT	238	0.340	0.283	0.403	5.193	15.256

Continued on next page

Table 5: Full per-condition results under the unified 238-example protocol (continued).

Dataset	Model	Strategy	N	Accuracy	CI Low	CI High	Latency (s)	VRAM (GB)
ARC-Challenge	Gemma-4-26B-A4B	Few-shot CoT	238	0.937	0.899	0.961	2.765	48.067
ARC-Challenge	Gemma-4-E2B	Few-shot CoT	238	0.727	0.667	0.780	4.485	9.543
ARC-Challenge	Gemma-4-E4B	Few-shot CoT	238	0.891	0.845	0.924	0.831	14.895
ARC-Challenge	Phi-4-Mini-Reasoning	Few-shot CoT	238	0.290	0.236	0.351	4.088	7.145
ARC-Challenge	Phi-4-Reasoning	Few-shot CoT	238	0.277	0.224	0.337	4.854	27.305
ARC-Challenge	Qwen3-30B-A3B	Few-shot CoT	238	0.487	0.425	0.551	9.651	57.621
ARC-Challenge	Qwen3-8B	Few-shot CoT	238	0.525	0.462	0.588	4.889	15.256
ARC-Challenge	Gemma-4-26B-A4B	Zero-shot	238	0.945	0.909	0.968	3.433	48.067
ARC-Challenge	Gemma-4-E2B	Zero-shot	238	0.769	0.711	0.818	1.379	9.543
ARC-Challenge	Gemma-4-E4B	Zero-shot	238	0.899	0.854	0.931	0.154	14.895
ARC-Challenge	Phi-4-Mini-Reasoning	Zero-shot	238	0.273	0.220	0.333	4.089	7.145
ARC-Challenge	Phi-4-Reasoning	Zero-shot	238	0.303	0.248	0.364	4.853	27.305
ARC-Challenge	Qwen3-30B-A3B	Zero-shot	238	0.416	0.355	0.479	9.605	57.621
ARC-Challenge	Qwen3-8B	Zero-shot	238	0.445	0.384	0.509	4.897	15.256
GSM8K	Gemma-4-26B-A4B	CoT	238	0.782	0.725	0.829	11.617	48.067
GSM8K	Gemma-4-E2B	CoT	238	0.685	0.623	0.741	5.015	9.543
GSM8K	Gemma-4-E4B	CoT	238	0.790	0.734	0.837	5.076	14.895
GSM8K	Phi-4-Mini-Reasoning	CoT	238	0.210	0.163	0.266	8.433	7.145
GSM8K	Phi-4-Reasoning	CoT	238	0.008	0.002	0.030	9.810	27.305
GSM8K	Qwen3-30B-A3B	CoT	238	0.756	0.698	0.806	14.415	57.621
GSM8K	Qwen3-8B	CoT	238	0.790	0.734	0.837	6.871	15.256
GSM8K	Gemma-4-26B-A4B	Few-shot CoT	238	0.681	0.619	0.737	14.346	48.067
GSM8K	Gemma-4-E2B	Few-shot CoT	238	0.710	0.649	0.764	4.515	9.543
GSM8K	Gemma-4-E4B	Few-shot CoT	238	0.752	0.694	0.803	5.405	14.895

Continued on next page

Table 5: Full per-condition results under the unified 238-example protocol (continued).

Dataset	Model	Strategy	N	Accuracy	CI Low	CI High	Latency (s)	VRAM (GB)
GSM8K	Phi-4-Mini-Reasoning	Few-shot CoT	238	0.315	0.259	0.377	8.153	7.145
GSM8K	Phi-4-Reasoning	Few-shot CoT	238	0.021	0.009	0.048	9.660	27.305
GSM8K	Qwen3-30B-A3B	Few-shot CoT	238	0.807	0.752	0.852	14.740	57.621
GSM8K	Qwen3-8B	Few-shot CoT	238	0.819	0.765	0.863	6.674	15.256
GSM8K	Gemma-4-26B-A4B	Zero-shot	238	0.794	0.738	0.841	8.921	48.067
GSM8K	Gemma-4-E2B	Zero-shot	238	0.693	0.632	0.748	4.549	9.543
GSM8K	Gemma-4-E4B	Zero-shot	238	0.790	0.734	0.837	4.931	14.895
GSM8K	Phi-4-Mini-Reasoning	Zero-shot	238	0.181	0.137	0.235	8.328	7.145
GSM8K	Phi-4-Reasoning	Zero-shot	238	0.042	0.023	0.076	9.731	27.305
GSM8K	Qwen3-30B-A3B	Zero-shot	238	0.782	0.725	0.829	14.302	57.621
GSM8K	Qwen3-8B	Zero-shot	238	0.790	0.734	0.837	6.664	15.256
MATH L1–L3	Gemma-4-26B-A4B	CoT	238	0.643	0.580	0.701	17.412	48.067
MATH L1–L3	Gemma-4-E2B	CoT	238	0.580	0.516	0.641	10.964	9.543
MATH L1–L3	Gemma-4-E4B	CoT	238	0.668	0.606	0.725	10.833	14.895
MATH L1–L3	Phi-4-Mini-Reasoning	CoT	238	0.223	0.174	0.280	16.378	7.145
MATH L1–L3	Phi-4-Reasoning	CoT	238	0.013	0.004	0.036	19.318	27.305
MATH L1–L3	Qwen3-30B-A3B	CoT	238	0.576	0.512	0.637	26.157	57.621
MATH L1–L3	Qwen3-8B	CoT	238	0.277	0.224	0.337	13.506	15.256
MATH L1–L3	Gemma-4-26B-A4B	Few-shot CoT	238	0.571	0.508	0.633	23.528	48.067
MATH L1–L3	Gemma-4-E2B	Few-shot CoT	238	0.689	0.628	0.744	8.969	9.543
MATH L1–L3	Gemma-4-E4B	Few-shot CoT	238	0.693	0.632	0.748	8.278	14.895
MATH L1–L3	Phi-4-Mini-Reasoning	Few-shot CoT	238	0.391	0.331	0.454	16.543	7.145
MATH L1–L3	Phi-4-Reasoning	Few-shot CoT	238	0.029	0.014	0.059	19.389	27.305
MATH L1–L3	Qwen3-30B-A3B	Few-shot CoT	238	0.613	0.550	0.673	24.786	57.621

Continued on next page

Table 5: Full per-condition results under the unified 238-example protocol (continued).

Dataset	Model	Strategy	N	Accuracy	CI Low	CI High	Latency (s)	VRAM (GB)
MATH L1–L3	Qwen3-8B	Few-shot CoT	238	0.639	0.576	0.697	11.515	15.256
MATH L1–L3	Gemma-4-26B-A4B	Zero-shot	238	0.693	0.632	0.748	14.125	48.067
MATH L1–L3	Gemma-4-E2B	Zero-shot	238	0.618	0.555	0.677	11.851	9.543
MATH L1–L3	Gemma-4-E4B	Zero-shot	238	0.630	0.567	0.689	11.952	14.895
MATH L1–L3	Phi-4-Mini-Reasoning	Zero-shot	238	0.252	0.201	0.311	16.289	7.145
MATH L1–L3	Phi-4-Reasoning	Zero-shot	238	0.038	0.020	0.070	19.265	27.305
MATH L1–L3	Qwen3-30B-A3B	Zero-shot	238	0.584	0.521	0.645	27.392	57.621
MATH L1–L3	Qwen3-8B	Zero-shot	238	0.218	0.171	0.275	13.270	15.256
TruthfulQA MC1	Gemma-4-26B-A4B	CoT	238	0.727	0.667	0.780	2.698	48.067
TruthfulQA MC1	Gemma-4-E2B	CoT	238	0.563	0.500	0.625	1.273	9.543
TruthfulQA MC1	Gemma-4-E4B	CoT	238	0.643	0.580	0.701	2.476	14.895
TruthfulQA MC1	Phi-4-Mini-Reasoning	CoT	238	0.983	0.958	0.993	4.092	7.145
TruthfulQA MC1	Phi-4-Reasoning	CoT	238	0.996	0.977	0.999	4.600	27.305
TruthfulQA MC1	Qwen3-30B-A3B	CoT	238	0.987	0.964	0.996	9.961	57.621
TruthfulQA MC1	Qwen3-8B	CoT	238	0.987	0.964	0.996	5.156	15.256
TruthfulQA MC1	Gemma-4-26B-A4B	Few-shot CoT	238	0.786	0.729	0.833	1.681	48.067
TruthfulQA MC1	Gemma-4-E2B	Few-shot CoT	238	0.517	0.454	0.580	6.020	9.543
TruthfulQA MC1	Gemma-4-E4B	Few-shot CoT	238	0.739	0.680	0.791	0.195	14.895
TruthfulQA MC1	Phi-4-Mini-Reasoning	Few-shot CoT	238	0.966	0.935	0.983	4.143	7.145
TruthfulQA MC1	Phi-4-Reasoning	Few-shot CoT	238	1.000	0.984	1.000	4.362	27.305
TruthfulQA MC1	Qwen3-30B-A3B	Few-shot CoT	238	0.966	0.935	0.983	9.625	57.621
TruthfulQA MC1	Qwen3-8B	Few-shot CoT	238	0.979	0.952	0.991	5.021	15.256
TruthfulQA MC1	Gemma-4-26B-A4B	Zero-shot	238	0.794	0.738	0.841	2.653	48.067

Continued on next page

Table 5: Full per-condition results under the unified 238-example protocol (continued).

Dataset	Model	Strategy	N	Accuracy	CI Low	CI High	Latency (s)	VRAM (GB)
TruthfulQA MC1	Gemma-4-E2B	Zero-shot	238	0.550	0.487	0.612	2.613	9.543
TruthfulQA MC1	Gemma-4-E4B	Zero-shot	238	0.735	0.676	0.787	0.456	14.895
TruthfulQA MC1	Phi-4-Mini-Reasoning	Zero-shot	238	0.945	0.909	0.968	4.129	7.145
TruthfulQA MC1	Phi-4-Reasoning	Zero-shot	238	0.996	0.977	0.999	4.403	27.305
TruthfulQA MC1	Qwen3-30B-A3B	Zero-shot	238	0.983	0.958	0.993	9.907	57.621
TruthfulQA MC1	Qwen3-8B	Zero-shot	238	0.962	0.930	0.980	4.586	15.256

D Additional Statistical Results

This appendix reports supplemental statistical tables supporting the main deployment-aware interpretation. Table 6 reports bootstrap confidence intervals for the strongest weighted configurations. Table 7 reports paired permutation comparisons among the strongest weighted configurations. Table 8 reports a compact model-level summary of weighted rank instability across prompting strategies.

Table 6: Bootstrap summary for the six strongest weighted configurations under the unified 238-example protocol.

Model	Strategy	Point estimate	Bootstrap mean	95% CI low	95% CI high	Bootstrap std.
Gemma-4-26B-A4B	Zero-shot	0.794	0.794	0.766	0.822	0.0146
Gemma-4-E4B	Few-shot CoT	0.761	0.761	0.731	0.792	0.0154
Gemma-4-E4B	CoT	0.759	0.759	0.728	0.788	0.0150
Gemma-4-E4B	Zero-shot	0.758	0.759	0.728	0.788	0.0153
Gemma-4-26B-A4B	CoT	0.756	0.756	0.726	0.784	0.0149
Qwen3-8B	Few-shot CoT	0.722	0.723	0.693	0.753	0.0152

Table 7: Paired permutation comparisons among the strongest weighted configurations. The weighted leader remains significantly ahead of the strongest Gemma-4-E4B operating points under the current weighted metric, while the differences among Gemma-4-E4B prompting variants are negligible.

Model A	Strategy A	Model B	Strategy B	Weighted delta	Two-sided p
Gemma-4-26B-A4B	Zero-shot	Gemma-4-E4B	Few-shot CoT	0.033	0.0238
Gemma-4-26B-A4B	Zero-shot	Gemma-4-E4B	CoT	0.035	0.0160
Gemma-4-26B-A4B	Zero-shot	Gemma-4-E4B	Zero-shot	0.036	0.0150
Gemma-4-E4B	Few-shot CoT	Gemma-4-E4B	CoT	0.002	0.8912
Gemma-4-E4B	Few-shot CoT	Gemma-4-E4B	Zero-shot	0.003	0.8690
Gemma-4-E4B	CoT	Gemma-4-E4B	Zero-shot	0.000	0.9784

Table 8: Compact prompt-instability summary across weighted rankings. *Weighted rank range* is the difference between the best and worst weighted rank of a model across CoT, few-shot CoT, and zero-shot prompting. *Dataset best-strategy flips* indicates whether the model's best prompting strategy changes across the four benchmark datasets.

Model	Rank (CoT)	Rank (Few-shot CoT)	Rank (Zero-shot)	Weighted rank range	Dataset best strategy flips
Qwen3-8B	5	2	5	3	0
Gemma-4-E2B	3	5	3	2	1
Gemma-4-26B-A4B	2	3	1	2	1
Gemma-4-E4B	1	1	2	1	0
Phi-4-Mini-Reasoning	6	6	6	0	1
Phi-4-Reasoning	7	7	7	0	0
Qwen3-30B-A3B	4	4	4	0	1

Table 9: Deployment-budget and efficiency summary under the unified 238-example protocol. The upper block reports the best weighted configuration under selected VRAM budgets. The lower block reports the strongest configurations by the combined efficiency score weighted accuracy/(latency × VRAM).

Summary type	Model	Strategy	Weighted acc.	Latency (s)	VRAM (GB)
Best under ≤16 GB	Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895
Best under ≤24 GB	Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895
Best under ≤48 GB	Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895
Best unrestricted	Gemma-4-26B-A4B	Zero-shot	0.794	7.283	48.067
Top efficiency score	Gemma-4-E2B	CoT	0.658	4.425	9.543
Second efficiency score	Gemma-4-E4B	Few-shot CoT	0.761	3.677	14.895
Third efficiency score	Gemma-4-E2B	Zero-shot	0.671	5.098	9.543

Table 10: Compatibility diagnostics for Phi-4-Reasoning under the unified evaluation pipeline. Missing-prediction rate corresponds to the share of cases with no scoreable extracted prediction. Malformed-output rate is reported for the math-style tasks where the expected final-answer format is explicit.

Dataset	Strategy	Accuracy	Missing pred. rate	Think tag rate	Malformed output rate
ARC-Challenge	CoT	0.294	0.000	0.992	–
ARC-Challenge	Few-shot CoT	0.277	0.000	1.000	–
ARC-Challenge	Zero-shot	0.303	0.000	0.950	–
GSM8K	CoT	0.008	0.983	1.000	0.139
GSM8K	Few-shot CoT	0.021	0.975	0.992	0.059
GSM8K	Zero-shot	0.042	0.958	0.975	0.155
MATH L1–L3	CoT	0.013	0.000	0.962	0.122
MATH L1–L3	Few-shot CoT	0.029	0.000	0.954	0.101
MATH L1–L3	Zero-shot	0.038	0.000	0.924	0.122
TruthfulQA MC1	CoT	0.996	0.000	0.950	–
TruthfulQA MC1	Few-shot CoT	1.000	0.000	1.000	–
TruthfulQA MC1	Zero-shot	0.996	0.000	0.916	–