Accuracy and performance analysis

1) Run context and test objectives

As part of our LLM inference optimization work, we conducted a test campaign to compare three models along two complementary dimensions:

1. **Inference performance**: throughput (tokens/s) and latency (including TTFT and percentile latencies).
2. **Model quality / accuracy**: results on a set of standard evaluation benchmarks (zero-shot / few-shot depending on the configuration).

The goal is not only to identify "the fastest model", but to quantify the performance vs quality trade-off and determine the most suitable model depending on the target use case (general-purpose production, reasoning/math, strict latency constraints, etc.).

2) Evaluated models

The following models were evaluated:

- gpt-oss-120b
- qwen3-235b
- llama4-scout-17b-16e

3) Accuracy / quality metrics (benchmark-based)

Model quality was evaluated using five standard benchmarks:

- arc_challenge
- gsm8k
- hellaswag
- truthfulqa_mc2
- winogrande

Each benchmark produces a task-specific score

1) ARC-Challenge

- **Typical metric:** normalized accuracy.
- **What it measures:** multiple-choice question answering requiring reasoning over scientific facts and commonsense.
- **How to interpret:** higher is better. "Normalized" scoring is commonly used to reduce bias from answer choice length.

2) GSM8K

- **Typical metric:** Exact Match.
- **What it measures:** grade-school math word problems, requires multi-step reasoning and arithmetic.
- **How to interpret:** higher is better.

3) HellaSwag

- **Typical metric:** normalized accuracy
- **What it measures:** commonsense reasoning / selecting the most plausible continuation of a situation.
- **How to interpret:** higher is better. Normalization helps reduce length bias between choices.

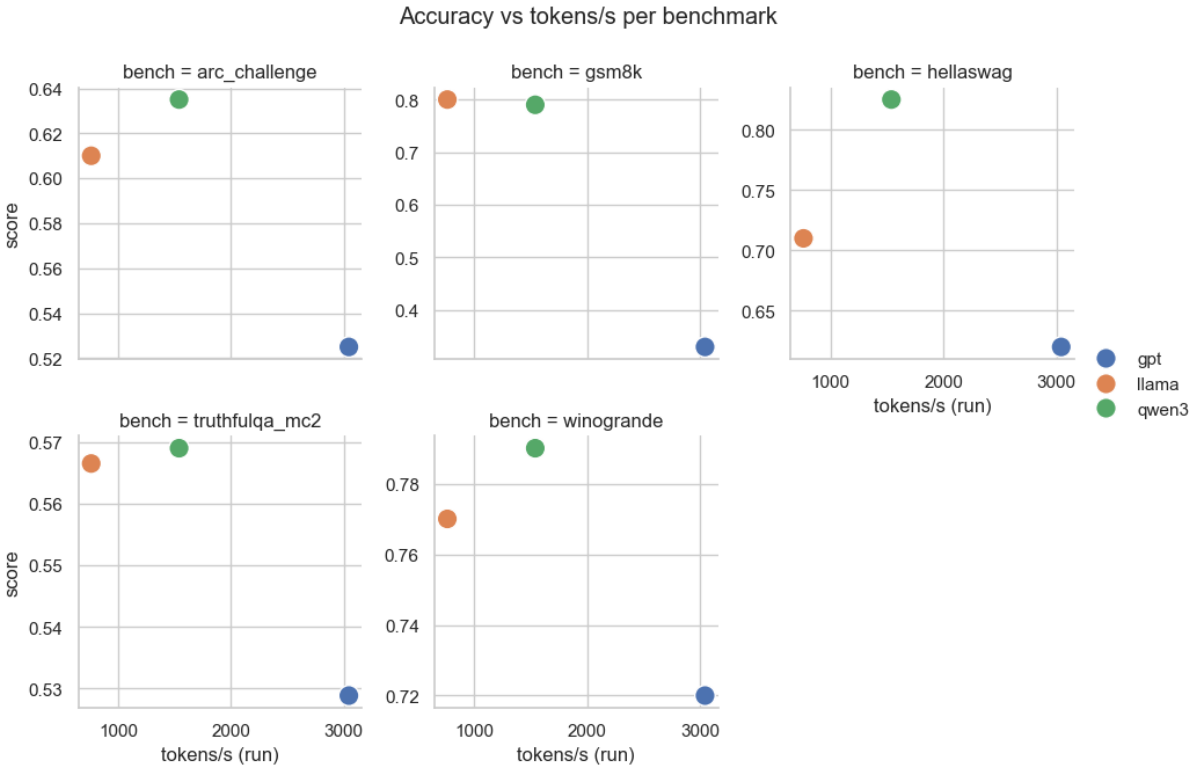
4) TruthfulQA MC2

- **Metric: MC2** score.
- **What it measures:** truthfulness / resistance to common misconceptions in a multiple-choice setting.
- **How to interpret:** higher is better. This benchmark is especially relevant for checking whether a model confidently selects misleading answers.

5) WinoGrande

- **Typical metric:** accuracy
- **What it measures:** pronoun/coreference resolution requiring commonsense reasoning (who or what does the pronoun refer to?).
- **How to interpret:** higher is better.

4) Figure 1: Accuracy vs Throughput



Key observations by task:

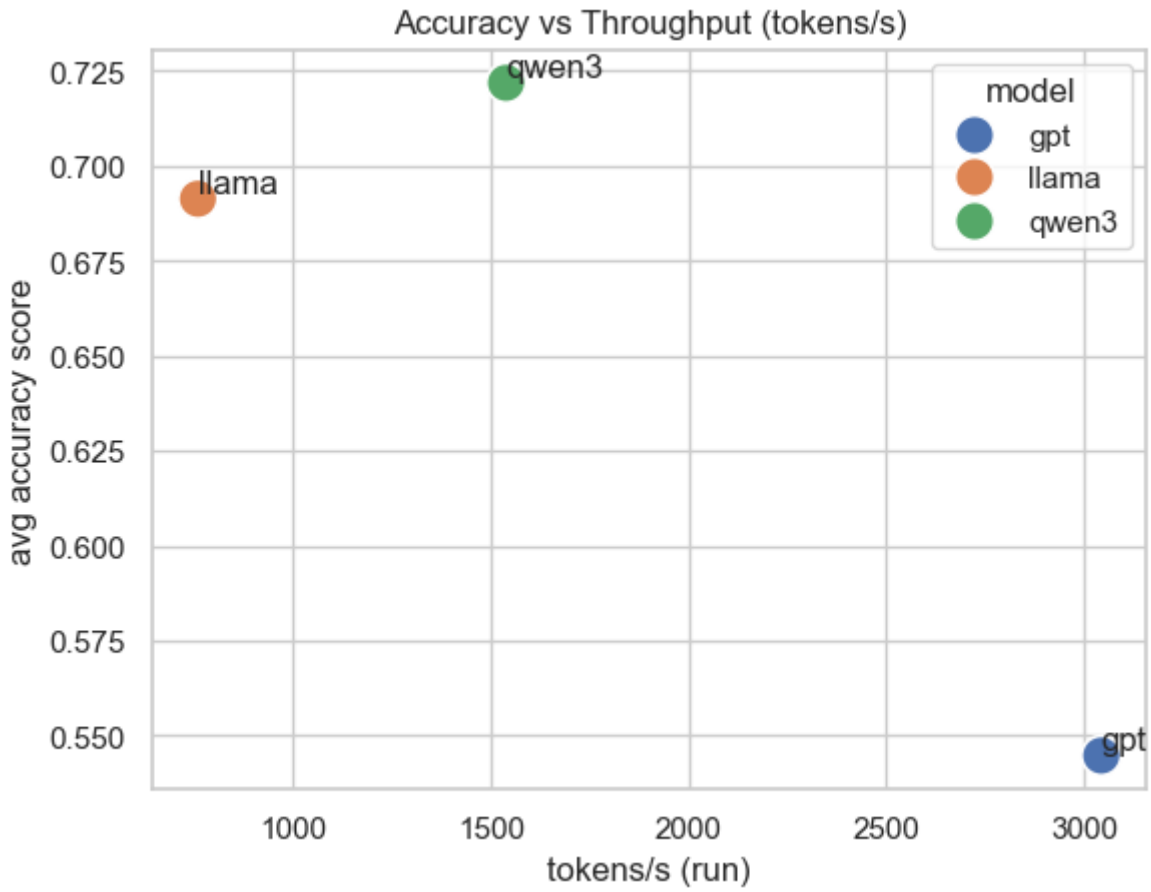
- **ARC-Challenge:** qwen3 (0.635) > llama (0.610) > gpt (0.525)
- **GSM8K:** llama (0.800) ≳ qwen3 (0.790) >> gpt (0.330)
- **HellaSwag:** qwen3 (0.825) > llama (0.710) > gpt (0.620)
- **TruthfulQA MC2:** qwen3 (0.569) ≳ llama (0.566) > gpt (0.529)
- **WinoGrande:** qwen3 (0.790) > llama (0.770) > gpt (0.720)

Interpretation:

- **qwen3 is the strongest generalist**: best on 4/5 tasks.
- **llama is best on GSM8K** (math/reasoning) but loses elsewhere and is slower.
- **gpt is the fastest** but consistently the weakest on accuracy, especially on GSM8K.

If we care about overall quality, qwen3 dominates; if our workload is heavily math-centric, llama may still be worth considering despite lower throughput.

Figure 2) Accuracy vs Throughput (tokens/s)



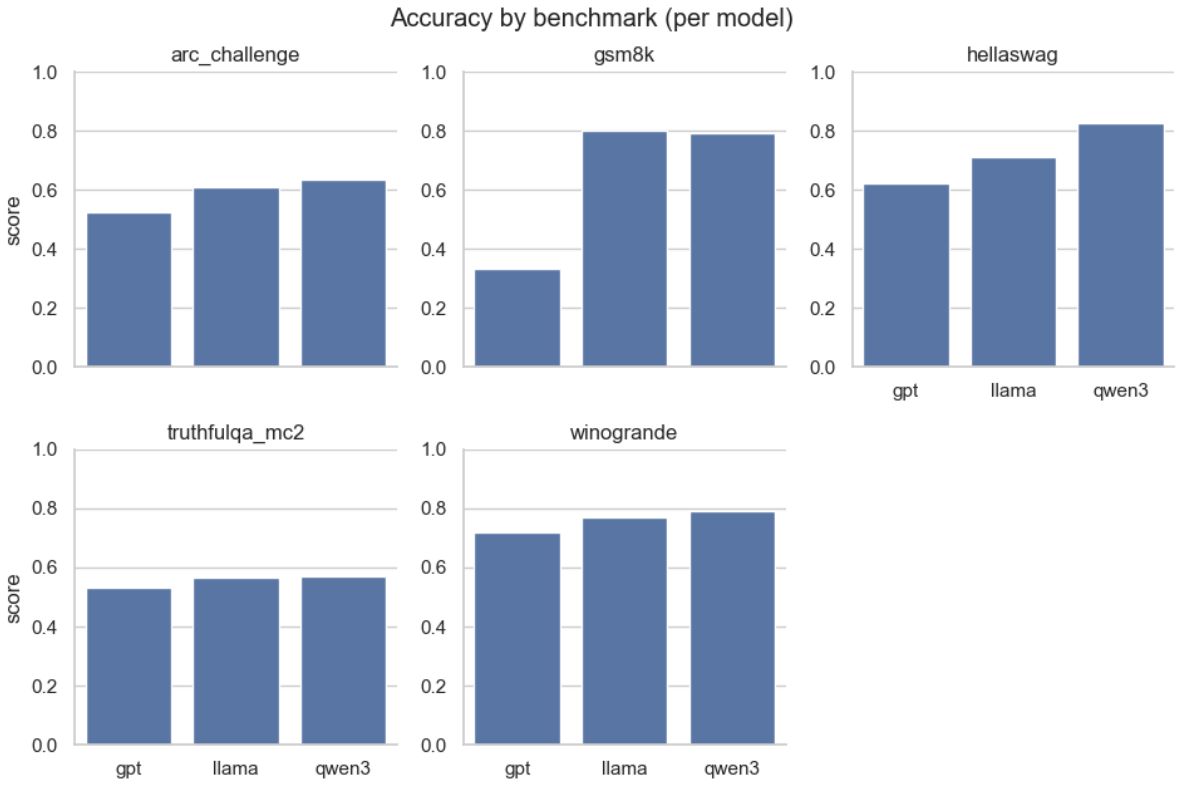
- **gpt:** avg ≈ **0.545**, throughput ≈ **3050 tok/s**
- **llama:** avg ≈ **0.691**, throughput ≈ **760 tok/s**
- **qwen3:** avg ≈ **0.722**, throughput ≈ **1550 tok/s**

Interpretation:

- **qwen3 is the best balance** (highest accuracy + decent throughput).
- **gpt is throughput-first** with a significant quality penalty.
- **llama is Pareto-dominated by qwen3** on the aggregate view (qwen3 is both faster and more accurate overall), even though llama wins on GSM8K specifically.

For a general-purpose deployment target, qwen3 is the best candidate under this setup.

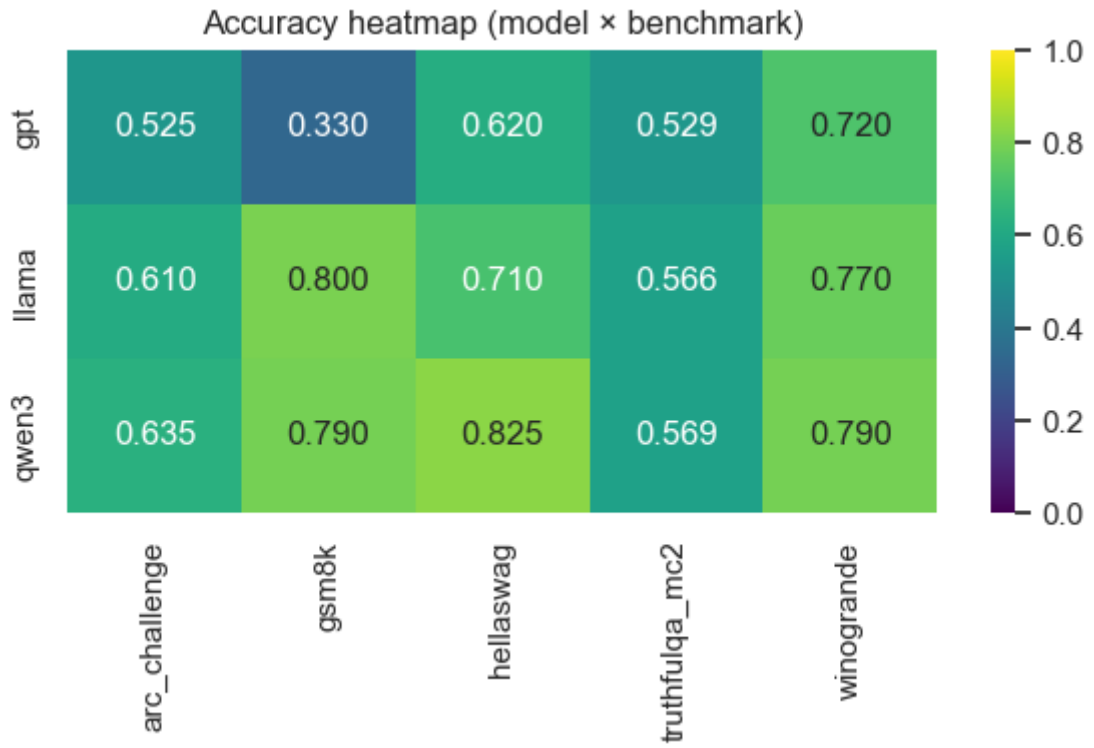
Figure 3) Accuracy by benchmark (per model)



Interpretation:

- Matches Figure 1: qwen3 leads most tasks; llama leads GSM8K; gpt trails.

Figure 4) Accuracy heatmap



Interpretation:

- GSM8K is the largest gap: gpt (0.33) vs llama/qwen3 (~0.8).
- qwen3 is consistently high across tasks; llama is strong but less consistent.

Conclusion: Quality vs speed and best-fit workloads

Overall takeaway

The three models exhibit a clear quality–throughput trade-off. qwen3 provides the best overall balance (highest average accuracy with strong throughput), gpt maximizes throughput at a notable quality cost, and llama stands out primarily on math/reasoning workloads.

Model-by-model summary:

qwen3: Best overall quality / best general-purpose trade-off

- **Speed:** mid-high throughput (~1.55k tokens/s).
- **Quality:** highest average accuracy (~0.72) and top scores on 4 out of 5 benchmarks.
- **Where it's best:**
    - **General knowledge + commonsense** prompts ("choose the most plausible continuation", everyday reasoning) strong on **HellaSwag** (0.825).
    - **Science/logic multiple-choice QA** prompts, best on **ARC-Challenge** (0.635).
    - **Coreference / pronoun resolution** and structured commonsense, best on **WinoGrande** (0.790).
    - **Truthfulness / avoiding misconceptions** in MC format, slightly best on TruthfulQA MC2 (0.569).
- **Recommended use:** default model for production when you need **consistent quality across diverse prompts** without sacrificing too much throughput.

Llama: Best for math and step-by-step reasoning (but slower)

- **Speed:** lowest throughput (~0.76k tokens/s).
- **Quality:** strong overall (~0.69 average), but less consistent than qwen3 across tasks.
- **Where it's best:**

    - **Math word problems and multi-step arithmetic reasoning**, best on **GSM8K** (0.800), narrowly ahead of qwen3.
    - Prompts that demand explicit computation, “show your steps”, numerical constraints, or strict final answers.

- **Trade-off:** we gain reasoning accuracy (especially math) but pay in throughput (capacity/cost).
- **Recommended use:** we can pick llama when our workload is dominated by math/quantitative reasoning and latency/throughput constraints are less strict.

Gpt: Fastest throughput, lowest quality

- **Speed:** highest throughput (~3.05k tokens/s).
- **Quality:** lowest average accuracy (~0.55), with a major weakness on math (**GSM8K 0.33**).
- **Where it’s best:**
    - **Low-stakes**, high-volume tasks where speed matters more than correctness, such as:
        - short, templated text generation,
        - summarization where occasional errors are acceptable,
        - lightweight classification or routing,
        - draft generation that is later reviewed/filtered.
- **Recommended use:** throughput-oriented scenarios where we can tolerate lower accuracy or add safeguards (post-checking, reranking, fallback model).