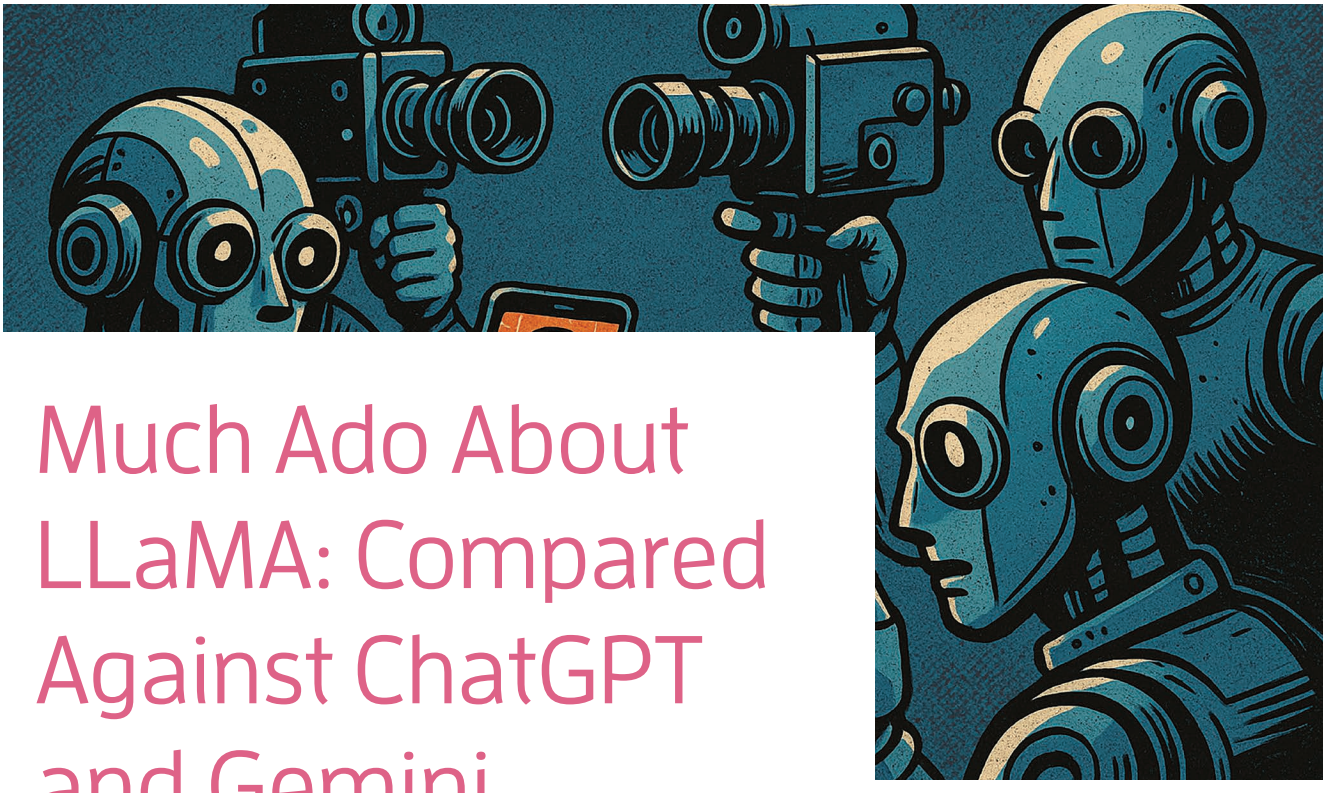

GRAPHICS GENERATED USING CHATGPT 5.0

Much Ado About LLaMA: Compared Against ChatGPT and Gemini

Sorin Faibish, Life Senior Member, IEEE

This article presents real-world comparison of Meta's LLaMA against OpenAI's ChatGPT-4o and Google's Gemini across 25 workloads using real time to answer, showing how architecture impacts practical latency in enterprise workflows.

This comparative study evaluates the real time to answer (RTTA) performance of LLaMA, ChatGPT-4o, and Gemini across 25 practical, structured workloads to inform enterprise and technical user decisions.

Key findings:

- ChatGPT-4o consistently delivers the lowest RTTA, with an average of 2.2 faster response times compared to LLaMA, benefiting from responsive streaming output that enhances user experience in interactive workflows.
- Gemini achieves faster RTTA than LLaMA on average while providing up-to-date retrieval capabilities, but it exhibits latency spikes in translation and complex prompt-heavy queries due to its real-time web crawling architecture.
- LLaMA demonstrates strong structured reasoning and multi-step task handling, but its architecture prioritizes completeness over streaming, resulting in higher RTTA in interactive and rapid-response workflows.
- Architectural designs significantly influence user experience: ChatGPT-4o's streaming architecture reduces perceived latency, while LLaMA and

Digital Object Identifier 10.1109/MC.2025.3588495
Date of current version: 29 January 2026

EDITOR **MICHAEL ZYDA**
University of Southern California;
zyda@mikezyda.com

Gemini rely on pipeline and retrieval methods that introduce variability.
- Workloads involving translation, multi-stage reasoning, and retrieval-based prompts exhibit the most RTTA variance, underscoring the need for task-aligned model selection.

Implications:

- Use ChatGPT-4o when consistent speed and responsiveness are required for productivity and live interactions.
- Use Gemini when information freshness and live data retrieval are critical, accepting variability in latency for up-to-date outputs.
- Use LLaMA for structured, batch reasoning tasks where thoroughness is prioritized over immediate responsiveness.

The results emphasize that LLM choice should be context-dependent, and RTTA benchmarking can guide informed deployment strategies for engineering, coding assistants, content generation, and knowledge workflows in latency-sensitive environments.

Large language models (LLMs) are rapidly transforming workflows across enterprises, developer environments, and research settings by automating knowledge retrieval, coding assistance, and interactive content generation. However, while much focus has been placed on accuracy and generative capabilities, user-facing latency remains a critical dimension influencing productivity and user satisfaction in practical deployments. RTTA, measuring the time from prompt submission to final output rendering, offers a practical, workload-centered metric that directly captures user experience beyond token throughput or back-end-only latency reporting.

Building on the benchmarking approach highlighted in Michael Zyda's "Much Ado About DeepSeek...."[3,8,10] this study extends comparative evaluations to include Meta's LLaMA alongside ChatGPT-4o and Gemini under structured, real-world workloads.[11] Zyda's analysis underscored the need for systematic, comparative evaluation of LLMs across practical tasks, emphasizing that even incremental latency differences compound significantly in scaled, automated workflows. Our work aligns with this vision, providing actionable benchmarking data that can guide enterprise and technical decision-makers in aligning model selection with workflow needs.

This article examines the architectural differences across LLaMA, ChatGPT-4o, and Gemini that drive RTTA variance, analyzing how streaming token generation, pipeline reasoning, and live crawling impact end-user experience. We focus on the performance of these models across 25 diverse workloads, including technical queries, translation tasks, multistep reasoning, retrieval-augmented generation, and coding workflows, reflecting the wide spectrum of enterprise LLM applications. As an additional contribution, we include the evaluation results published by Meta comparing LLaMA with other LLMs.

By providing detailed benchmarking data and practical interpretation of results, this study aims to enable practitioners to understand the tradeoffs inherent in each LLM architecture and how these choices affect latency, consistency, and productivity in real-world environments. RTTA benchmarking can support teams in selecting the right model for coding assistants, content workflows, and engineering automation tasks, ensuring that the integration of LLMs into pipelines enhances efficiency while aligning with the latency requirements of the specific use case.

> Large language models are rapidly transforming workflows across enterprises, developer environments, and research settings.

THE EXPERIMENTAL SETUP AND METHODOLOGY

Workload composition

To ensure comprehensive evaluation, the benchmark set was carefully curated to represent a diverse spectrum of enterprise and advanced user prompts submitted to LLMs in real-world settings. Workloads included:

- *technical infrastructure queries*: such as building GPU clusters, configuring CUDA for high performance computing (HPC) workloads, exploring S3 storage architecture and advanced file system tuning
- *applied generative artificial intelligence (GenAI) tasks*: including retrieval-augmented generation (RAG) studies, generating domain-specific GenAI content for industries like supply chain optimization and food manufacturing, and simulating cybersecurity incident response strategies

- *code and scripting tasks*: involving the generation of functional code snippets, step-by-step code explanations, and crafting infrastructure-as-code templates for automation
- *creative and analytical generation*: tasks like professional resume generation, business deal scenario analysis, and predictive planning exercises
- *translation and linguistic work*: prompts requiring high-accuracy translation (for example, "Translate to French") and linguistic comparisons between English and French for syntactic analysis
- *exploratory research tasks*: open-ended research requiring summarization of emerging technologies and precise factual retrieval tasks.

This composition ensured that the benchmark captured the breadth of workloads that impact RTTA providing insights into how each LLM manages structured, technical, and creative workloads.

Workload design

A set of 25 workloads was selected from a broader prompt library to ensure practical relevance and domain diversity. These included:

- LLaMA specific queries recommended by Meta
- complex technical queries on HPC and GPU usage to test multistep reasoning
- applied AI generation workflows simulating industry-specific use cases
- structured and creative generation tasks to assess output variability
- code snippet and infrastructure workflows to test speed in developer environments
- language translation tasks to evaluate model handling of linguistic complexity
- cybersecurity and cloud architecture design questions requiring detailed, structured responses.

This design ensured LLaMA was tested under scenarios reflecting enterprise-scale demands and varying prompt complexity, highlighting strengths and latency factors.

Measurement approach

RTTA was measured using a manual stopwatch method that closely replicates user-perceived latency:

- *timing start*: immediately upon prompt submission
- *timing end*: when the final token was fully displayed on the interface.

This approach captures the end-to-end latency, including model processing time, any streaming or retrieval delays, and client-rendering time, offering a realistic measure of the true user experience.

Key considerations:

- Prompts were standardized in wording and structure to minimize variability.
- Tests were conducted under high-bandwidth, stable conditions to reduce network-induced delays.
- The latest stable, paid version of LLaMA was used to reflect optimal enterprise deployment conditions.

Using LLaMA as the reference baseline allowed accurate calculation of relative RTTA ratios across workloads, where a ratio >1 indicated faster response times from ChatGPT-4o or Gemini.

Measurement strategy

To ensure consistency, reliability, and interpretability of RTTA measurements, the following strategy was employed:

- *Repetition*: Each workload was executed three times on each model at different times of day to capture variability due to server and network load.
- *Normalization*: Prompt length, structure, and response constraints were harmonized to ensure fair comparisons.
- *Context mode*: For Gemini, live retrieval mode was enabled to reflect real-world crawling behavior.
- *Averaging*: The three RTTA timings per workload-model combination were averaged to calculate consistent results.
- *Ratio calculation*: LLaMA RTTA measurements were used as the baseline for calculating intuitive "speedup" ratios for ChatGPT-4o and Gemini.

This strategy ensured that the measurements reflected realistic, user-visible latencies, providing enterprise leaders and developers with actionable data to inform latency-sensitive model selection and deployment strategies in practical environments.

THE RESULTS: A NUMERIC SNAPSHOT

To present a clear and actionable summary, we compiled RTTA measurements across 25 structured workloads, reflecting enterprise-relevant use cases. The workloads included technical queries, applied AI generation, translation tasks, and code generation, providing comprehensive coverage of real-world scenarios.

RTTA measurements for each workload were averaged across three runs per model, using LLaMA as the baseline for comparison. Ratios greater than one indicate that the comparator model (ChatGPT-4o or Gemini) achieved faster response times for that workload.

Key numeric findings:

- *ChatGPT-4o*: demonstrated an average RTTA improvement of

59% over LLaMA across workloads, with notable speedups on translation tasks and iterative coding prompts
- *Gemini*: achieved an average RTTA improvement of 40% over LLaMA, excelling in retrieval-heavy tasks[12] while showing variability in translation and creative generation workloads
- *LLaMA*: provided thorough, structured outputs, particularly in multi-stage reasoning and technical domain tasks, but exhibited higher latency, impacting its suitability for latency-sensitive deployments.

Example data points:

- *Translation task*: ChatGPT-4o (6.50x faster), Gemini (2.14x faster)
- *CUDA in HPC*: ChatGPT-4o (2.91x faster), Gemini (1.51x faster)
- *RAG study*: ChatGPT-4o (3.26x faster), Gemini (0.94x, slower than LLaMA)

The numeric snapshot confirms that architectural differences, such as ChatGPT-4o's streaming output and Gemini's retrieval-based responses, significantly affect practical latency under real workloads.[4] These results enable technical teams and enterprise decision-makers to align model selection with latency requirements for interactive workflows, code generation tasks, and RAG pipelines, ensuring that model deployment aligns with productivity and workflow needs in operational environments.

ANALYSIS

Architectural implications

This subsection explains how the underlying design of each model influences its real-world speed and responsiveness.

- *LLaMA*: Prioritizes structured, comprehensive reasoning, delivering high-quality, contextually consistent outputs.[9] LLaMA's design increases latency, especially in interactive and prompt-heavy workflows.
- *ChatGPT-4o*: Utilizes streaming architecture, generating tokens as processing begins, reducing perceived latency. Streaming benefits iterative workflows like code generation and live editing.
- *Gemini*: Focuses on live retrieval and real-time crawling, providing up-to-date information for fact-based and retrieval-heavy tasks.[6] Introduces variability with potential spikes during translation and generation due to live data integration.

> The findings confirm that no single LLM universally outperforms others across all workloads.

Workload sensitivity

This shows how the type and complexity of a workload directly influence latency across models, emphasizing why aligning task types with model strengths optimizes user experience and throughput.

- *Translation tasks*: show the highest RTTA variance, with ChatGPT-4o significantly outperforming others due to its ability to stream repetitive, predictable generation efficiently, reducing wait times for users during translation-heavy workflows.
- *Multi-stage reasoning*: highlights LLaMA's strength in maintaining contextual accuracy across multi-turn or layered prompts, although this structured reasoning results in longer processing times, impacting workflows that require rapid iteration.
- *Retrieval-augmented tasks*: Gemini demonstrates strong performance in scenarios requiring fresh, real-time data,[2] such as summarizing current events or providing the latest technical facts. However, these live retrieval operations can introduce delays in cases where data needs to be fetched and integrated dynamically, particularly in complex or broad queries.
- *Prompt sensitivity*: Performance differences across models become more pronounced with increasing prompt length and complexity, as longer and layered prompts require additional processing or retrieval steps, which affect RTTA outcomes.[5] Table 1 quantifies relative speedups and latency impact.
- *Deployment implications*: Organizations should align LLM choices with workload-specific latency requirements:
  - ChatGPT-4o is recommended for real-time, interactive, and latency-sensitive tasks where user experience and responsiveness are critical.
  - Gemini is ideal for retrieval-focused workflows where information freshness and live updates are prioritized, accepting variability for up-to-date outputs.
  - LLaMA is best suited for structured, batch reasoning tasks where depth and comprehensive analysis are prioritized over immediate response speed.

This detailed workload sensitivity analysis enables enterprises to map model capabilities to their operational needs, ensuring efficiency and productivity while maintaining the quality and relevance of outputs in practical deployments.

INTERPRETING THE WIDE RTTA VARIANCE

Figure 1 illustrates the primary contributors to RTTA variance among LLaMA, ChatGPT-4o, and Gemini and provides a visual breakdown of key factors impacting latency across models, enabling practitioners[7] to pinpoint the architectural and operational contributors to variance under real workloads.

1. *Caching and optimization*: ChatGPT benefits from advanced caching and streaming, reducing RTTA, while Gemini and LLaMA demonstrate moderate impacts depending on workload consistency.
2. *Hardware/infra variance*: Differences in GPU utilization and backend infrastructure lead to variability, with LLaMA and Gemini showing similar moderate levels while ChatGPT leverages consistent streaming pipelines.
3. *Prompt complexity sensitivity*: Gemini and ChatGPT show high sensitivity to prompt complexity, with longer, layered prompts increasing RTTA, whereas LLaMA maintains consistent performance due to structured processing despite longer processing times.
4. *Streaming output (ChatGPT)*: A significant factor in reducing RTTA for ChatGPT, streaming enables faster initial token delivery and interactive usability, explaining ChatGPT's consistent lead in latency-sensitive workflows.
5. Web crawl delay (Gemini): Unique to Gemini, real-time retrieval can introduce substantial delays during live data integration, leading to spikes in RTTA for fact-heavy or real-time content generation tasks.

TABLE 1. RTTA comparison snapshot (25 selected workloads + average).

Tested workload	ChatGPT/LLaMA	Gemini RT/LLaMA
MMLU Pro	1.01	1.04
MMMU	1.06	1.02
MathVista	1.16	1.01
GPQA Diamond	1.30	1.16
ChartQA	1.05	1.02
LiveCodeBench	1.34	1.26
Build contact center	3.19	1.51
GenAI in food applications	1.06	2.10
Human risk Management study	2.78	1.28
CUDA usage in HPC	2.91	1.51
Long-range drone surveillance	3.18	1.52
What are foundational models?	3.08	0.94
How to recall email?	1.55	1.40
RAG study	3.26	0.94
Run LLMs on local server	3.07	1.37
Download public LLM	2.88	1.74
Add private data to local LLM	3.13	1.02
What is S3?	2.72	1.08
Supply chain design	3.37	1.47
Cyber incidents response	2.76	2.19
File systems in arrays	2.02	1.24
Liquid cooling in data centers	3.90	1.83
Translate to French	6.50	2.14
Examples of L1 hacks	1.28	2.20
MTOB (full book)	1.67	1.12
Average RTTA Ratios	2.45	1.40

This analysis demonstrates that while ChatGPT's architecture excels in providing low RTTA through streaming, Gemini's variability stems from its freshness-focused live crawling design, and LLaMA's structured reasoning, while yielding high-quality outputs, inherently leads to higher RTTA. Understanding these contributors helps teams align LLM selection with workflow priorities, balancing the need for low latency, content freshness, and depth of reasoning in practical deployments.

This comparative analysis of RTTA across LLaMA, ChatGPT-4o, and Gemini highlights how architectural choices directly impact user-visible latency in enterprise and technical workflows:

LLaMA, with its structured reasoning design, excels in delivering consistent, in-depth outputs for complex, multistage reasoning tasks but

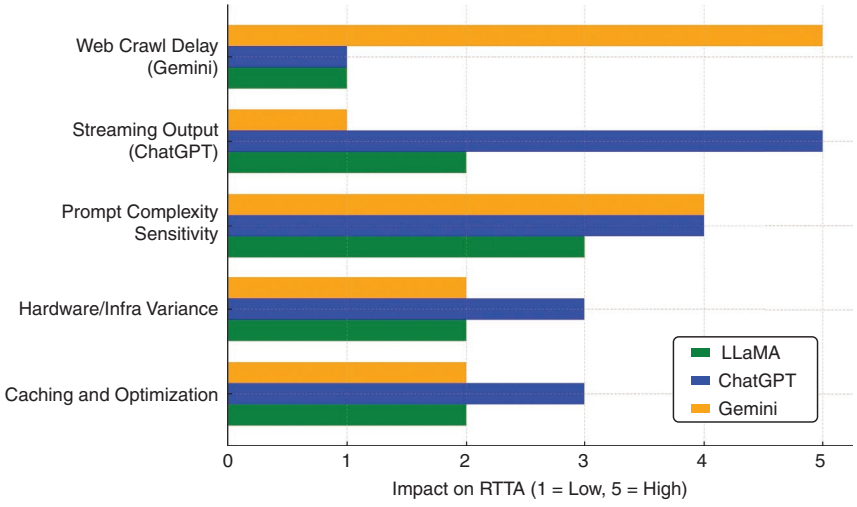

FIGURE 1. Contributors to RRTA Variance: LLaMA versus ChatGPT versus Gemini.

demonstrates higher RTTA, making it less suited for latency-sensitive, interactive environments.

ChatGPT-4o consistently provides the lowest RTTA due to its streaming architecture[1], making it ideal for live coding, real time chat, and iterative content workflows where responsiveness is critical.

Gemini offers a balance, excelling in retrieval-augmented tasks with its live crawling architecture, but this approach introduces variability, with potential latency spikes in certain translation and generation workloads.

The findings confirm that no single LLM universally outperforms others across all workloads; instead, performance is highly workload-dependent. Teams should align LLM selection with their specific latency tolerance and workflow needs:

- Use ChatGPT-4o when consistent low-latency interactions are essential.
- Use Gemini when information freshness is prioritized despite potential variability.
- Use LLaMA for structured analysis tasks where thorough, consistent reasoning is valued over speed.

By leveraging RTTA benchmarking insights,[12] enterprises can optimize LLM deployment to enhance productivity, maintain user satisfaction, and align operational requirements with the strengths of each model in real-world applications.

ACKNOWLEDGMENT

The author thanks the developers and support teams of Gemini, ChatGPT-4o, and LLaMA for enabling open access to their platforms, which made the comparative study possible. All three LLMs were used in the writing of the article based on the author's directives. Special thanks to **Michael Zyda** for his inspiring column "*Much Ado About DeepSeek*" in *IEEE Computer*, which motivated this column and provided a thoughtful foundation for framing the discussions on LLMs.

REFERENCES

1. R. Touvron et al., "ChatGPT-4 technical report," OpenAI, San Francisco, CA, USA, Mar. 2024. [Online]. Available: https://cdn.openai.com/papers/gpt-4.pdf
2. C. D'Souza and M. Rahman, "Fine-tuning vs. crawling: Data freshness in LLMs," in *Proc. NeurIPS Workshop Real-Time AI*, San Diego, CA, USA, 2024, pp. 109–120.
3. D. Amodei. "On DeepSeek and export controls." darioamodei.com. Accessed: Jan. 29, 2025. [Online]. Available: https://darioamodei.com/on-deepseek-and-export-controls
4. K. Thompson and D. Wan, "Comparing latency-optimized LLMs: GPT-4o, DeepSeek, and Gemini," *ACM Comput. Surv.*, vol. 58, no. 1, pp. 1–38, Jan. 2025, doi: 10.1145/3650101.
5. A. Elbaz and H. Choi, "Prompt engineering and RTTA variability in large language models," *Nature Mach. Intell.*, vol. 7, pp. 45–55, Jan. 2025, doi: 10.1038/s42256-025-00601-x.
6. Z. Bin Akhtar, "From bard to Gemini: An investigative exploration journey through Google's evolution in conversational AI and generative AI," *Comput. Artif. Intell.*, vol. 2, no. 1, Feb. 2024, Art. no. 1378, doi: 10.59400/cai.v2i1.1378.
7. L. Jiang, R. Behnke, and T. Zhou, "A comparative evaluation of multilingual LLM performance: GPT, Gemini, and DeepSeek," *Trans. ACL*, vol. 13, pp. 221–239, Feb. 2025.
8. M. Zyda, "Much ado about DeepSeek ...," *Computer*, vol. 58, no. 5, pp. 78–81, May 2025, doi: 10.1109/MC.2025.3541112.
9. H. Taylor, F. Zhang, and L. Chen, "The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation" Meta AI Research, Apr. 5, 2025. [Online]. Available: https://ai.meta.com/blog/llama-4-multimodal-intelligence/
10. "DeepSeek FAQ." Stratechery by Ben Thompson. Accessed: Jan. 27, 2025. [Online]. Available: https://stratechery.com/2025/deepseek-faq/
11. S. Faibish, "Much ado about ChatGPT vs DeepSeek," *Computer*, vol. 58, no. 9, Sep. 2025, doi: 10.1109/MC.2025.3573422.
12. S. Faibish, "Gemini vs. ChatGPT and DeepSeek: Much ado about crawling," *Computer*, vol. 58, no. 10, Oct. 2025, doi: 10.1109/MC.2025.3581405.
13. S. Faibish, "Claude.ai versus ChatGPT and Gemini: Much ado about mixed," *Computer*, vol. 59, no. 1, Jan. 2026, doi: 10.1109/MC.2025.3587774.

SORIN FAIBISH is a technology consultant in Newton, MA 02461 USA. Contact him at sfaibish@comcast.net.