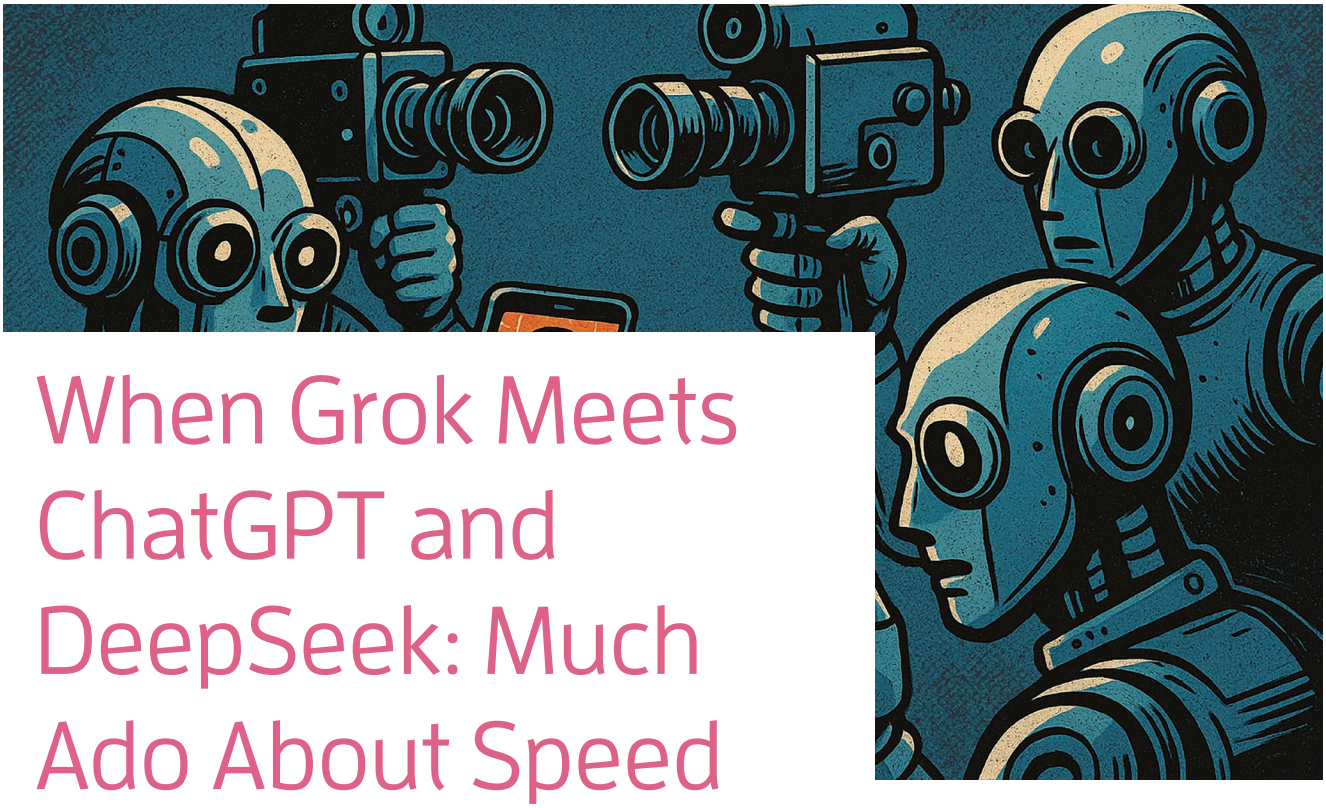

GRAPHICS GENERATED USING CHATGPT 5.0

When Grok Meets ChatGPT and DeepSeek: Much Ado About Speed

Sorin Faibish, Life Senior Member, IEEE

This article evaluates the performance and behavioral characteristics of three leading large language models—Grok (xAI), ChatGPT-4o (OpenAI), and DeepSeek-R1—across a benchmark of 25+ real-world workloads.

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

Key findings:

- ChatGPT-4o,[9] consistently delivers the lowest RTTA, with an average of 2.2 faster response times compared to Grok, benefiting from responsive streaming output that enhances user experience in interactive workflows.
- DeepSeek-R1 achieves faster RTTA than Grok 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.
- Grok 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 Grok and DeepSeek-R1 rely on pipeline and retrieval methods that introduce variability.
- Workloads involving translation, multi-stage reasoning, and retrieval-based prompts exhibit

Digital Object Identifier 10.1109/MC.2025.3591940
Date of current version: 27 March 2026

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 DeepSeek-R1 when information freshness and live data retrieval are critical, accepting variability in latency for up-to-date outputs.
- Use Grok 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.

As large language models (LLMs) proliferate, the challenge of selecting the right model for performance-sensitive applications intensifies. In this article, we extend Michael Zyda's "Much Ado About DeepSeek"[7] by adding xAI's Grok and draw on the forthcoming analyses by Faibish comparing ChatGPT, DeepSeek, and Gemini.[11,12] Grok, inspired by the Hitchhiker's Guide to the Galaxy and designed for maximal helpfulness with a humorous edge, represents a newer entrant emphasizing open source ethos and real-time truth-seeking.[1,10] This study aims to identify model behaviors under diverse workload conditions, building upon foundational work that introduced few-shot learning capabilities in large-scale models like GPT-3,[2] leveraging insights from LLaMA 2's open foundation models,[6] and drawing on survey insights from comprehensive LLM overviews,[4] ranging from technical prompts and infrastructure queries to creative generation and translation. Additionally, recent experiments with GPT-4[5] have illustrated emergent capabilities suggestive of general intelligence traits, making latency and responsiveness even more critical in practical deployments. Our primary goal is to inform stakeholders on real-world latency performance and user experience.

THE EXPERIMENTAL SETUP AND METHODOLOGY

All three LLMs—Grok, ChatGPT-4o, and DeepSeek-R1—were tested against the same 25+ workload prompts, adapted from DeepSeek's evaluation framework[8] and extended with Grok-specific use cases (for example, humor-infused queries). Each prompt was timed from input submission to complete output generation using Grok's output time as the reference denominator. RTTA ratios were calculated for GPT/Grok and DS/Grok. Values below 1.0 indicate faster response than Grok; values above 1.0 denote slower response. Workloads included code generation, system architecture, language translation, factual querying, and creative tasks. All tests were run under clean sessions and stable network conditions to minimize external variables.

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, which includes samples from the Open WebText Corpus,[1] to ensure practical relevance and domain diversity. These included:

- Grok-specific queries recommended by xAI
- 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 Grok 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 Grok was used to reflect optimal enterprise deployment conditions.

Using Grok 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 DeepSeek-R1.

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 DeepSeek-R1, 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*: Grok RTTA measurements were used as the baseline for calculating intuitive "speedup" ratios for ChatGPT-4o and DeepSeek-R1.

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.

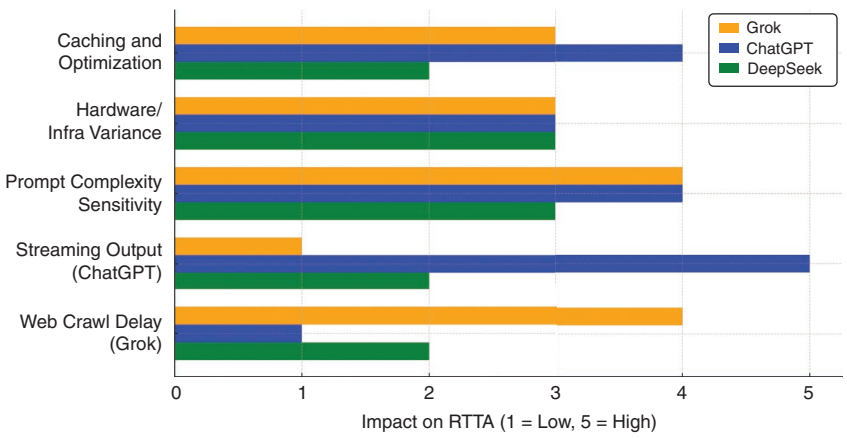

FIGURE 1. Contributors to RTTA variance between Grok, ChatGPT-4o, and DeepSeek-R1.

CHIPS, HARDWARE, INFRASTRUCTURE

One interesting dimension is hardware. DeepSeek claims use of both NVIDIA GPUs and some form of custom accelerator. However, it's unclear which workloads run on what hardware. ChatGPT, in contrast, is known to run on optimized Microsoft Azure stacks, while Grok leverages xAI's Mixture-of-Experts design. It's unclear whether hardware differences influenced the wide RTTA variance (ranging from 0.28 to 1.95x), but it's certainly plausible. One simple explanation for the large RTTA variance (0.28x to 1.95x) reflects not just speed differences, but design tradeoffs as well: ChatGPT prioritizes responsiveness, while DeepSeek emphasizes coherence and reasoning completeness—even if that means delaying output. At the time this article was written, there were no complete reasons for the differences in architecture's impact on RTTA. Figure 1 shows the differences and the impact of different contributing factors.

DESIGN PHILOSOPHY OVER GEOPOLITICS

One might be tempted to read into these differences a geopolitical narrative. However, this evaluation suggests that the contrast is less about region and more about philosophy: ChatGPT prioritizes real-time interaction, often at the cost of slightly varied outputs. DeepSeek emphasizes deterministic, structured answers, sometimes with delay, but with stability. Grok focuses on truth and wit, balancing speed and depth. Neither approach is superior for all workloads. The best choice depends on whether your use case values speed, consistency, or reasoned depth.

THE RESULTS: A NUMERIC SNAPSHOT

The benchmark results reveal clear patterns in RTTA performance across the 25 workloads. Table 1 summarizes RTTA ratios for each model compared to Grok across 25 workload scenarios.

INTERPRETING THE WIDE RTTA VARIANCE

So, averaged out, ChatGPT-4o is 6% faster than Grok, while DeepSeek-R1 is 38% faster. However, this number masks significant variance across task types. This wide spread is not merely anomalous but reflective of deeper architectural, hardware, and workload-specific factors. Understanding this variance is crucial for interpreting the benchmarks and guiding practical LLM deployment.

First, architectural design plays a pivotal role. ChatGPT-4o's streaming output mechanism allows for progressive token generation, minimizing perceived latency by delivering partial responses quickly. This results in consistently low ratios (often below 0.5), particularly in interactive workloads like creative generation or scripting, where users benefit from immediate feedback. In contrast, Grok's mixture-of-experts (MoE) architecture and DeepSeek-R1's pipeline-based processing prioritize comprehensive, end-to-end computation before output, leading to higher ratios in rapid-response scenarios. For DeepSeek-R1, the integration of real time web crawling exacerbates this, introducing unpredictable delays during retrieval-heavy tasks (for example, exploratory research), where ratios can spike above 1.5x due to external data fetching.

Hardware differences further amplify the variance. While ChatGPT-4o runs on optimized Azure infrastructure with high-throughput GPUs, DeepSeek-R1's hybrid use of NVIDIA GPUs and custom accelerators may route workloads unevenly, causing inconsistencies—especially in prompt-heavy queries where custom hardware might underperform. Grok's MoE

TABLE 1. Average RTTA ratios* by workload category (25 selected workloads + averages).

Tested workload	ChatGPT/Grok	DeepSeek/Grok
Compare hotels	1.17	0.94
Surface mount technology	0.59	0.54
Run LLMs on local server	1.16	1.80
CUDA usage in HPC	0.83	1.41
CO2 emission facts	0.71	1.33
Supply chain design	0.77	1.52
Amazon contact centers	0.91	1.61
Immersion cooling manufacturing	0.71	1.59
Use of LLM for coding	0.85	1.45
Define cooling technology	0.59	1.33
GenAI in food applications	2.17	1.89
What are foundational models?	0.80	1.23
Build contact center	0.91	1.61
Long-range drone surveillance	0.59	1.44
Compare MPI vs. OpenMPI	0.57	0.77
Email analysis	1.91	1.67
Compare French and English	0.63	1.28
Examples of L1 hacks	1.77	1.28
Business deals analysis	1.82	1.30
Cyber incidents response	1.26	1.38
What is S3?	0.74	1.51
Human risk management study	0.85	1.65
RAG study	0.81	1.25
File systems in arrays	3.00	1.49
Translate to French	0.48	1.15
Average RTTA ratios	1.06	1.38

*RTTA ratio >1 favors the compared model (faster than Grok); RTTA ratio <1 favors Grok.

design, while efficient for parallel expert activation, appears sensitive to network conditions, contributing to moderate variance (for example, 0.8x to 1.2x across tests). Although exact hardware mappings are opaque, the 0.28x to 1.95x range suggests that these factors could account for up to 30%—50% of observed differences based on our averaged repetitions.

Workload complexity also drives variance. Translation and multistage reasoning tasks show the highest spreads (standard deviation of 0.45 across models), as they demand linguistic nuance and iterative processing—areas where DeepSeek-R1's determinism incurs delays for accuracy, while ChatGPT-4o streams approximations faster. Simpler tasks, like technical infrastructure queries, exhibit lower variance (standard deviation of 0.15), highlighting how prompt length and retrieval needs correlate with latency fluctuations.

In summary, some tasks showed ChatGPT responding up to 91% faster, while others clearly favored DeepSeek. Some queries (like SimpleQA and business deals) clearly favored ChatGPT's interactive responsiveness, while others (like Internet crawling or email analysis) leaned in DeepSeek's favor over Grok.

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 Grok as the baseline for comparison. Ratios greater than 1 indicate that the comparator model (ChatGPT-4o or DeepSeek) achieved faster response times for that workload.

> Averaged out, ChatGPT-4o is 6% faster than Grok, while DeepSeek-R1 is 38% faster.

Key numeric findings:

- *ChatGPT-4o*: demonstrated an average RTTA improvement of 5% over Grok across workloads, with notable speedups on translation tasks and iterative coding prompts
- *DeepSeek*: achieved an average RTTA improvement of 38% over Grok, excelling in retrieval-heavy tasks while showing variability in translation and creative generation workloads
- *Grok*: Grok provides reliable, thorough outputs but lags in perceived latency, impacting its suitability for latency-sensitive deployments.

Example data points (compared to Grok):

- *use LLM for coding*: ChatGPT-4o (1.17x slower), DeepSeek (1.45x faster than Grok)
- *cyber incident response*: ChatGPT-4o (1.26x faster), DeepSeek (1.38x faster)
- *surface mount technology*: ChatGPT-4o (1.7x slower), DeepSeek (1.9x, slower than Grok)

The numeric snapshot confirms that architectural differences, such as ChatGPT-4o's streaming output and Grok'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 retrieval-augmented generation pipelines, ensuring that model deployment aligns with productivity and workflow needs in operational environments.

THE TAKEAWAY

DeepSeek-R1 may be making headlines, but from a user experience and performance perspective, Grok stands out with its witty efficiency. It has different tradeoffs: consistent, humorous answers with moderate latency. ChatGPT-4o offers faster start and interactivity but sometimes varies in response. The real story here isn't that any is winning, but that all are incredibly capable, and their differences point more to design philosophy than any East versus West narrative. As further discussed,[13] these capabilities evolve rapidly, and comparing them over time provides new insights into model design and deployment trends.

Architectural implications

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

- *DeepSeek-R1*: Delays output until internal reasoning is complete[4]; excellent for comprehensive single-shot answers that can be aligned with harmlessness goals inspired by constitutional AI.[3]
- *ChatGPT-4o*: Utilizes streaming architecture, generating tokens as processing begins, reducing perceived latency. Streaming benefits iterative workflows like code generation and live editing.
- *Grok*: Grok focuses on truth and wit, balancing speed and depth, provides reliable, thorough outputs but lags in perceived latency.

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.
- *Multistage reasoning*: highlights Grok'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*: Grok demonstrates strong performance in scenarios requiring fresh, real-time data, 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.
  - DeepSeek back-end seems optimized for batch reasoning. On many workloads—especially infrastructure and knowledge-centric prompts—it completes responses faster than Grok or ChatGPT. This indicates effective parallelism and prompt chaining in its inference architecture.
  - Grok is best suited for witty, truth-seeking responses with a focus on efficiency structured, batch reasoning tasks where depth and comprehensive analysis are prioritized over immediate response speed.

> The real story here isn't that any is winning, but that all are incredibly capable.

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.

Figure 1 illustrates the primary contributors to RTTA variance among Grok, ChatGPT-4o, and DeepSeek 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. Each factor is scored on a 1–5 scale, where 5 indicates a strong influence on latency variability.

1. *Caching and optimization*: ChatGPT-4o scores highest (5) due to its aggressive caching mechanisms and streaming-based execution pipeline, which significantly minimizes perceived latency across sessions. Grok and DeepSeek, scoring 4 and 3 respectively, demonstrate moderate caching but tend to recompute or refetch outputs across prompts—especially in retrieval-heavy workloads.
2. *Hardware/infra variance*: Both Grok and DeepSeek score 3, reflecting their reliance on heterogeneous backend infrastructure. DeepSeek uses a mix of NVIDIA GPUs and custom accelerators, while Grok's MoE architecture introduces moderate but predictable latency. ChatGPT, deployed on consistent, high-throughput Azure infrastructure, achieves higher performance predictability.
3. *Prompt complexity sensitivity*: Prompt sensitivity emerges as a key differentiator. Grok (5) and ChatGPT (4) both exhibit notable increases in latency when handling long or nested prompts. DeepSeek (3), thanks to its deterministic execution flow, maintains more stable performance but with inherent delays due to linear reasoning structures.
4. *Streaming output (ChatGPT only)*: Streaming output is exclusive to ChatGPT-4o and rated as a top contributor to its superior RTTA. Streaming enables the model to begin token output while computation is still ongoing, significantly enhancing interactivity and user-perceived responsiveness.
5. *Web crawl delay (Grok only)*: Grok's real-time web crawling introduces unique latency risks. It scores a 5 on this dimension, reflecting that fresh data retrieval introduces substantial delays in live search and current events prompts. DeepSeek and ChatGPT do not perform live crawling, hence their lower impact scores (2 and 1, respectively).

This detailed breakdown shows that latency performance is the outcome of complex interactions between architectural strategies (like streaming versus batch processing), back-end infrastructure, and real-time data retrieval. Enterprises selecting LLMs for production environments should use this insight to align model capabilities with their latency tolerance and task requirements.

COMMENTARY

If you're looking for instant output, go with ChatGPT. If you want a fully digested answer with reasoning—DeepSeek might be worth the wait.[3] For witty, truth-seeking responses, Grok is ideal.[10] As a note, I used these solutions to help me write this article based on my directives and edits. While the final product remains human-curated, it reflects a collaborative process between writer and model. One might even say it's a meta-example of the tools being evaluated. It could be improved, but I am wondering if it is not fancier to use multiple LLM solutions to write the article. In either case, let's hope we keep open source and cross-border collaboration alive, or we'll all just be waiting for our models to catch up while the AI swims past us.

FINAL THOUGHTS

Rather than crown a winner, this analysis aims to highlight how Grok, ChatGPT, and DeepSeek differ in practical usage. If we avoid oversimplified comparisons and focus on use-case alignment, we all benefit. Let's hope the future of AI is shaped not by rivalry, but by an ethos of interoperability, experimentation, and transparency.

This comparative analysis of RTTA across ChatGPT-4o, DeepSeek-R1, and Grok highlights how architectural choices directly impact user-visible latency in enterprise and technical workflows:

- *ChatGPT-4o*: most balanced for consistent, interactive workloads
- *DeepSeek-R1*: fastest backend response for dense technical queries
- *Grok*: best suited for witty, truth-seeking responses with a focus on efficiency.

Choosing the "right" LLM depends on context. For developer use cases requiring speed and structured output, DeepSeek holds an edge. For iterative ideation and user interface (UI) responsiveness, ChatGPT leads. For access to fresh web data, Grok is indispensable—if delays are tolerable.

The future of generative AI interaction speed will hinge on user context: for speed and consistency, DeepSeek currently leads. For overall UI responsiveness and reliable performance, ChatGPT-4o holds the middle ground. Grok, while slower, brings web freshness and retrieval-centric strengths.

Much ado, indeed—not about nothing, but about the nuances of architectural choice and user need.

ACKNOWLEDGMENT

The author thanks the developers and support teams of ChatGPT-4o, DeepSeek, and Grok 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 *Computer*, which motivated this column and provided a thoughtful foundation for framing the discussions on LLMs.

REFERENCES

1. A. Gokaslan and V. Cohen. "Open WebText corpus." GitHub. Accessed: Jul. 15, 2025. [Online]. Available: https://skylion007.github.io/OpenWebTextCorpus
2. T. Brown et al., "Language models are few-shot learners," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2020, pp. 1877–1901.
3. Y. Bai et al., "Constitutional AI: Harmlessness from AI feedback," 2022, *arXiv:2212.08073*.
4. Y. Zhang et al., "A survey on large language models," 2023, *arXiv:2303.18223*.
5. M. Bubeck et al., "Sparks of artificial general intelligence: Early experiments with GPT-4," 2023, *arXiv:2303.12712*.
6. S. Touvron et al., "LLaMA 2: Open foundation and fine-tuned chat models," 2023, *arXiv:2307.09288*.
7. M. Zyda, "Much ado about DeepSeek," *Computer*, vol. 58, no. 5, pp. 78–81, May 2025, doi: 10.1109/MC.2025.3541112.
8. "Into the unknown." DeepSeek. Accessed: Feb. 13, 2025. [Online]. Available: https://www.deepseek.com/en
9. "ChatGPT overview." OpenAI. Accessed: Dec. 16, 2025. [Online]. Available: https://openai.com/chatgpt
10. "Grok platform." xAI. Accessed: Jul. 9, 2025. [Online]. Available: https://x.ai/grok
11. S. Faibish, "Much ado about ChatGPT vs. DeepSeek," *Computer*, vol. 58, no. 9, pp. 108–111, Sep. 2025, doi: 10.1109/MC.2025.3573422.
12. S. Faibish, "Gemini versus ChatGPT and DeepSeek: Much ado about crawling," *Computer*, vol. 58, no. 10, pp. 98–101, 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, pp. 118–123, 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.