International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 24s, 2025
https://theaspd.com/index.php

The Evolution Of Large Language Models: A Comparative Survey Of Leading Approaches

Nirali Arora[1*], Dr Jaya Gupta[2], Priyanka Jeetendra Patil[3], Saylee Ameya Lapalikar[4], Poonam Nitin Tiware[5], Kanchan Girish Wankhede[6]

[1*]Asst Professor, CSE(AIML), AP Shah Institute of Technology Thane, Mumbai, Email: nrarora@apsit.edu.in
[2]Head of Department, CSE(AIML), AP Shah Institute of Technology Thane., Email: jdgupta@apsit.edu.in
[3]Assistant Professor, CSE-AIML, A.P. Shah Institute of Technology, Thane, Email-pjpatil@apsit.edu.in
[4]Assistant Professor, Information Technology, APSIT, Thane, Email: salapalikar@apsit.edu.in
[5]Assistant Professor, CSE-AIML, A.P. Shah Institute of Technology / Mumbai University, City- Thane, Email- pntiware@apsit.edu.in
[6]Assistant professor, CSE-AIML, A P Shah Institute of Technology, Thane (Mumbai),gwankhede@apsit.edu.in

Abstract

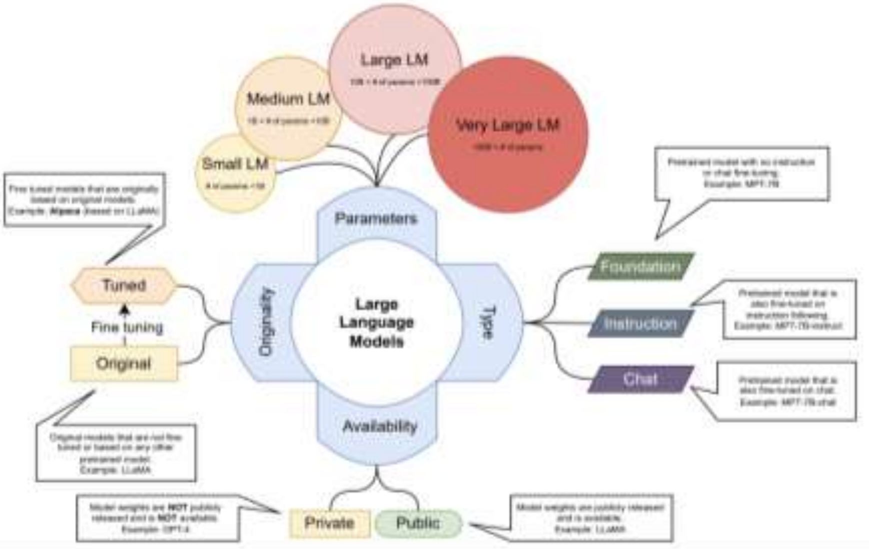
The exponential development of Large Language Models (LLMs) has significantly contributed to the evolution of the artificial intelligence (AI) landscape. This paper focuses on surveying several state-of-the-art LLMs, including big names in the industry like, OpenAI's GPT-4, Anthropic's Claude 3, Google's PaLM 2, and Meta's LLaMA 3.1. We evaluated these models based on their architecture, training data, performance and applications, using information from official papers, technical documentation and third-party reviews. By combining the claims from primary sources and third-party analysis, this survey hopes to give a summary of what the latest LLMs can do, what they can't and what's next for these cutting-edge domains of machine learning.

BACKGROUND

1. Technology Description

LLMs are fruits of one of the most rapidly evolving technologies in the realm of artificial intelligence and machine learning. These models often show great skill in understanding text and speech by using large, varied and complex datasets with strong computational power. In recent years many competitors in both development and research – from big players like OpenAI, Google and Anthropic to open-source groups like Meta and Stability AI – have turned the LLM landscape into a dynamic field of constant innovation. Besides their language skills, these models now join multimodal applications in more tech and social areas – allowing them to handle and create content in different forms like images, audio, video and text.

Figure reused from "Large Language Models: A Survey" by Minaee et al. (2024).[10]


Large LM
10B < # of params < 100B
Medium LM
1B < # of params < 10B
Very Large LM
100B < # of params
Small LM
# of params < 1B
Fine tuned models that are originally based on original models. Example: Alpaca (based on LLaMA)
Pretrained model with no instruction or chat fine-tuning. Example: MPT-7B
Parameters
Tuned
Fine tuning
Original
Originality
Large Language Models
Type
Foundation
Instruction
Chat
Pretrained model that is also fine-tuned on instruction following. Example: MPT-7B-instruct
Pretrained model that is also fine-tuned on chat. Example: MPT-7B-chat
Original models that are not fine tuned or based on any other pretrained model. Example: LLaMA
Availability
Model weights are NOT publicly released and is NOT available. Example: GPT-4
Private
Public
Model weights are publicly released and is available. Example: LLaMA

This diagram shows how Large Language Models (LLMs) are sorted using four primary attributes: parameters, type, originality, and availability. The first attribute parameter, categorizes LLMs by how big they are – counted by how many trainable parameters they have. Tiny models with less than 1 billion parameters, are light and fast but simple, making them challenging to use for complex jobs. Medium models with 1 to 10 billion parameters, find a middle ground between doing well and using less power. Large models, with 10 to 100 billion parameters, and very large models, with over 100 billion parameters, show great skill in understanding and making language – but need immense computational power to train and deploy. This sorting shows the trade-offs between size, performance and ease of use in making LLMs.

The second attribute, type, is more about the purpose and the fine-tuning of the model. Broadly speaking, LLMs can be classified into three types: foundation models, instruction-tuned models and chat models. Most foundation models are simply pre-trained on arbitrarily immense datasets without any task-specific finetuning, which allows them to be versatile — but probably not optimal for particular applications. Instruction-tuned models are further refined and tuned to follow instructions provided by the users better, thus making them able to perform tasks like summarization, question answering, etc. They are further fine-tuned on dialogue data for conversation tasks, which allows them to generate coherent and contextually appropriate responses in the context of conversation. These distinctions point to a growing trend of customizing LLMs for specific use cases while sharing a common foundation of pretrained knowledge.

The third attribute, originality, is a way of distinguishing original models from tuned derivatives. Original models are foundational systems designed from foundational principles (like LLaMA), which themselves can be the basis for subsequent fine-tuned models. Tuned models, such as Alpaca, are built on these originals via further training, optimizing them for specific tasks or domains. The ability to fine-tune such large models enables the research community to build upon existing models to approach new tasks more quickly and inexpensively. The quality of a tuned model is highly contingent on both the quality of the base model and the fine-tuning process, which highlights the importance of choosing a base model that is well-trained and strong.

Lastly, availability differentiates public from private models. Unlike the proprietary models, the weights of public models (e.g. LLaMA) are released to the public to let researchers, developers and organizations experiment, fine-tune or integrate the models into their applications. Private models like GPT-4, on the other hand, hold their weights confidential and limit the introduction of their model weights to a select audience that must use APIs or start licenses to associate with the model. Such differentiation mirrors their organizational strategies — public models foster transparency, collaboration, and innovation, whereas private models provide control, exclusivity, and monetization. The public vs private availability of LLMs also directly impacts the ecosystem of LLMs research and deployment, specifically accessibility and scalability.

There's been a substantial progress in model architectures and training methodologies which is driven by the need for scalable, efficient, and, importantly, accessible LLMs. Comparative models represent the state-of-the-art LLMs up to 2024. The list of models is diverse, including proprietary powerhouses and open-source contenders. The selection of each model is based on its unique contributions, performance, methodology or approach, and innovations.

This competition has driven innovation in a few key areas, including multimodality. Models like Gemini and PaLM have increased their capabilities from just understanding textual context to integrating image, video, and audio processing within their technical capabilities. This shift represents a fundamental evolution in AI structures - as it gives way to the development of several applications like real-time transcription, complex reasoning with visual aids, creative content generation, and medical diagnostic assistance.

This competition has fuelled breakthroughs in several key areas:

Multimodality: Models like Gemini and PaLM have expanded their capabilities beyond just understanding textual context. This was achieved by integrating image, video, and audio processing into their technical capabilities. This shift represents a fundamental evolution in AI structures as it gives way to applications like real-time transcription, complex reasoning with visual aids, creative content generation, and medical diagnostic assistance.

Scalability: Larger model sizes, such as Google's PaLM and Meta's LLaMA, with more than 340 and 400 billion parameters, respectively, are constantly pushing the boundaries of what LLMs can do in extracting context used for understanding and generating human-like text consecutively.

Accessibility: Open-source LLMs, such as Llama and Stable LM, give the opportunity for AI studies and research. The most advanced technologies are now made available to a much larger audience, creating an environment that is going to host further collaboration in the development of new applications of the field of artificial intelligence.

Efficiency: The smaller, optimized models, like Phi-2, demonstrate that architectural and data improvements can outstrip classic brute force scaling. These models provide great performance using fewer computational resources and are thus more sustainable and accessible for a variety of applications.

Task-Specific Adaptation: Specialized models, such as Gemma and Phi-2, are mostly attuned to optimizations toward particular use cases. With a focus on task-specific fine-tuning and the reduction of resources used, these models prove that bigger is not always better in AI.

Model Selection Reasoning

The chosen models are state-of-the-art large language models (LLMs) up to the year 2024. The selection includes both proprietary giants and competitive open-source alternatives, each chosen uniquely for its contribution, performance, methodologies, and innovations.

GPT-4

Although introduced earlier, GPT-4 still serves as the most popular model for reasoning and multitask learning. The latest updates of GPT-4 in 2024 only perfected its capabilities in the execution of complicated reasoning tasks while it consolidates its strong footholds in applications for commercial use. GPT-4 follows a Transformer-based architecture and is pre-trained on the prediction of subsequent tokens in text from publicly available data plus licensed third-party datasets. It was further fine-tuned through Reinforcement Learning from Human Feedback (RLHF).[1] OpenAI has not shared any details about the architecture, size, training computations, or anything related to the dataset construction of GPT-4, due to competitive and safety concerns. However, the following is known based on analyses and reports:

Model Size and Architecture

There is a wide guess that GPT-4 has 1.76 trillion parameters, although the number has not been explicitly confirmed by OpenAI. The estimates are based on benchmarking and other indirect evaluations.

GPT-4 is an extension of its predecessors' transformer architecture, but it's designed with multimodal capabilities that allow it to take both text and images as inputs. The model supports context windows up to 8,192 and 32,768 tokens—far above the limit for GPT-3.5.[b]

Training Dataset

OpenAI has not clearly stated the data sets used for training GPT-4. The datasets reportedly include a mixture of publicly available data and sources obtained through licensing; however, original sources are not disclosed.[a][b]

Training Methodology

Fine-tuning with RLHF has been performed at scale on GPT-4, which significantly increases its reliability and alignment; this also was a core component of previous models, including GPT-3.5 and ChatGPT.[a][b][1]

Capabilities

GPT-4 is at or near human levels on most professional and academic benchmarks. Most notably, it scored in the 90th percentile on a simulated Uniform Bar Examination.[c]

Figure reused from "Gpt-4 technical report." by Achiam et al. (2023).

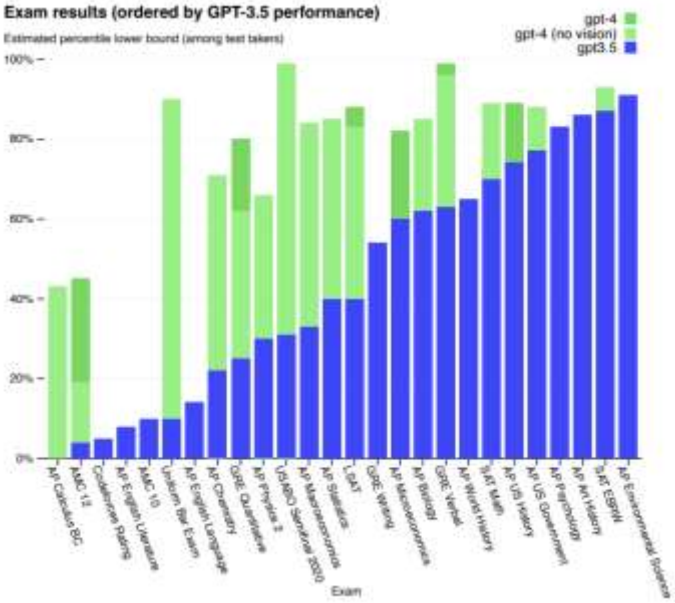


Figure 1.1 GPT exam results demonstrating human level accuracy

It scored highly on many standardized tests: SAT 1410 (94th percentile), LSAT 163 (88th percentile), Uniform Bar Exam 298 (90th percentile).[1] GPT-4 was able to pass specialized exams, among them those in oncology, engineering, and plastic surgery. On the Torrance Tests of Creative Thinking, it scored in the top 1% in originality and fluency, with flexibility scores ranging from the 93rd to the 99th percentiles.[1][a] Some research, however, questions the validity of certain measurements, like the Uniform Bar Exam.[1][b]

They show, for medical tasks, Microsoft researchers found that GPT-4 scored more than 20 points above the passing mark on the USMLE, outperforming its predecessor GPT-3.5 and even specialized models like Med-PaLM. However, they caution that LLMs, including GPT-4, carry extraordinary risks in medical applications because of the potential for inaccuracies and hallucinations. Other researchers from Columbia and Duke University showed that GPT-4 could help annotate cell types for the analysis of single-cell RNA-seq data.[b][3]

Limitations

GPT-4 keeps some of the problems inherent to its predecessors: it generates false or irrelevant information ("hallucinations"), and it's sometimes incorrect regarding reasoning. OpenAI warns users to exercise care with outputs for situations important to them through mechanisms like human reviews. GPT-4, however, shows quite a good improvement over GPT-3.5; hallucinations are reduced by 19 percentage points, while factual accuracy improves.[1]

Researchers at Microsoft tested GPT-4 on medical tasks and found that it exceeded the passing score for the USMLE by over 20 points, outperforming both earlier models like GPT-3.5 and models fine-tuned for medical knowledge, such as Med-PaLM. However, despite this strong performance, the report cautions that LLMs, including GPT-4, pose significant risks in medical applications due to potential inaccuracies and hallucinations. Additionally, researchers from Columbia and Duke University demonstrated GPT-4's ability to assist in cell type annotation in single-cell RNA-seq data analysis.[b][3]

Claude 3 Model Family

Anthropic's model family of Claude 3 includes three advanced models with increasing capabilities: Claude 3 Haiku, Claude 3 Sonnet, and Claude 3 Opus. All these models strongly emphasize alignment and safety, placing Anthropic at the forefront in ethical LLM development. Claude 3 introduces real-world reasoning and business integration improvements, enabling cheap fine-tuning while providing sophisticated alignment techniques. Each model in the family is tuned to different needs, so users can find the right balance between intelligence, speed, and cost for their particular applications.[c]

Claude 3 Haiku

The Claude 3 Haiku model is said to be outstanding in speed and cost efficiency. It can process dense information, such as a research paper filled with graphs and charts, in less than three seconds. That makes it perfect for applications where there needs to be fast answers, like knowledge retrieval and sales automation. The Claude Haiku also has enterprise-level security and reliability.

Anthropic has implemented a comprehensive suite of testing to minimize the risks of both toxic outputs and unauthorized access. The security framework in Haiku has been designed to monitor the system continuously, harden endpoints, apply secure coding practices, encrypt data strongly, and enforce strict access controls over sensitive information. Regular security audits have been performed and professional penetration testers have been hired to try and expose any vulnerabilities in its defenses.[e]

Claude 3 Sonnet

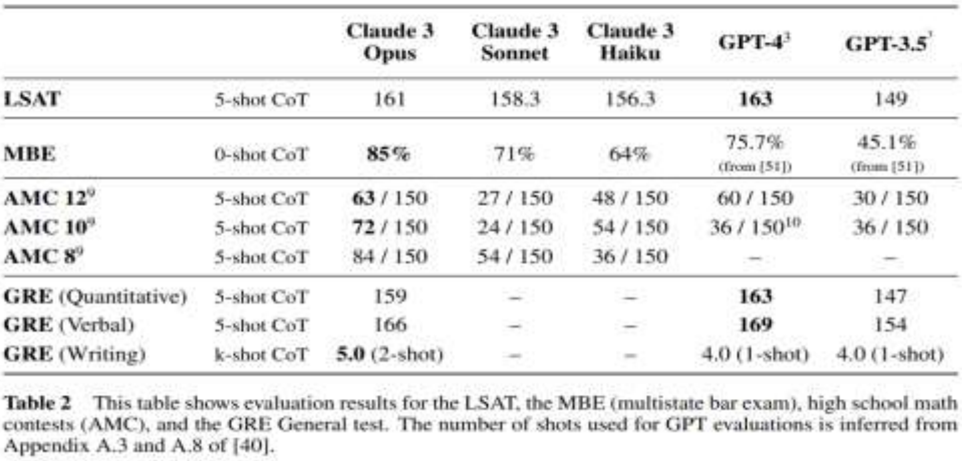
Claude 3 Sonnet is a big step up in speed and intelligence, twice as fast as Claude 2 and 2.1.[f] It excels in reasoning, sophisticated coding, and efficient computation, and it also retains a large context window of 200,000 tokens.[f]

Sonnet is a versatile platform in use in numerous applications from software engineering to high-order chatbot engagement, knowledge-based Q&A, visual data extraction, and robotic process automation. f Like Haiku, it boasts powerful security features like continuous monitoring of the system, secure coding practices, and access controls to safeguard any form of sensitive information.[f]

Claude 3 Opus

Claudette 3 Opus is at the top of the model family, showcasing state-of-the-art performance with a nearly human level of comprehension and fluency on highly complex tasks. The model outperforms all others in the standard AI testing benchmarks, including undergraduate-level expert knowledge (MMLU), graduate-level reasoning (GPQA), and basic math (GSM8K). Opus is especially well-suited for advanced applications, including complex analysis, subtle content creation, code generation, and good communication in non-English languages. Its prowess makes it a leader in the area of general intelligence improvements.[d] The development team has put into place rigorous testing protocols to ensure high safety standards, minimizing harmful outputs and unauthorized access.[d]

Figure reused from " The Claude 3 Model Family: Opus, Sonnet, Haiku." by Agrawal et al. (2023).



		Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4[3]	GPT-3.5[3]
LSAT	5-shot CoT	161	158.3	156.3	**163**	149
MBE	0-shot CoT	**85%**	71%	64%	75.7% (from [51])	45.1% (from [51])
AMC 12[9]	5-shot CoT	**63** / 150	27 / 150	48 / 150	60 / 150	30 / 150
AMC 10[9]	5-shot CoT	**72** / 150	24 / 150	54 / 150	36 / 150[10]	36 / 150
AMC 8[9]	5-shot CoT	84 / 150	54 / 150	36 / 150	–	–
GRE (Quantitative)	5-shot CoT	159	–	–	**163**	147
GRE (Verbal)	5-shot CoT	166	–	–	**169**	154
GRE (Writing)	k-shot CoT	**5.0** (2-shot)	–	–	4.0 (1-shot)	4.0 (1-shot)

Table 2 This table shows evaluation results for the LSAT, the MBE (multistate bar exam), high school math contests (AMC), and the GRE General test. The number of shots used for GPT evaluations is inferred from Appendix A.3 and A.8 of [40].

Figure 1.1 Claude vs GPT on standardized tests

Architecture and Model Size:

There are three models in the Claude 3 family: Claude 3 Opus, Claude 3 Sonnet, and Claude 3 Haiku, each tuned for a different level of capability. While the latter is the most powerful, Haiku is focused on speed and cost efficiency. Anthropic did not go into details regarding architecture or model size since these are kept proprietary.

Dataset Construction:

Anthropic hasn't released many details about the datasets it used to train the Claude 3 models. Most AI developers, like this one, likely use a combination of publicly available data, proprietary datasets, and curated data. The Claude 3 models were trained on a proprietary blend of internet data as of August 2023, third-party data, internally generated data, and contributions from data labeling services and contractors.[4]

Sophisticated data cleaning and filtering, deduplication, and classification methods ensure that the training inputs are of high quality. Most importantly, Claude 3 models have not been trained on any user input or output data originating from free users, Claude Pro users, or API customers.[4]

Training Method:

The Claude 3 models were heavily pre-trained on large datasets using methods such as word prediction in order to build up their language skills. Anthropic also used human feedback to encourage helpful, harmless, and honest responses.[4]A key feature of the training process of Claude 3 is the use of Constitutional AI; the models are aligned with human values through rules and principles derived from sources like the UN Declaration of Human Rights.[4] For Claude 3, an additional principle was added to underpin respect for disability rights, building on work in Collective Constitutional AI.[4]

Some of the human feedback data used in training was publicly released, along with research on reinforcement learning and red-teaming. Once trained, the models are put through rigorous safety evaluations. Anthropic's Trust and Safety team further uses continuous classifiers to monitor prompts and outputs for compliance with acceptable use policies and to minimize risks of harmful or malicious use.[4]

Gemini 1.5

The Gemini 1.5 model family is a significant advance in compute-efficient multimodal AI, capable of recalling and reasoning over millions of tokens of context. That includes processing lengthy documents, hours of video and audio, and demonstrating Google's vision for a fully multimodal AI system. It shows how text, visual, and audio tasks integrated into Gemini 1.5 raise the bar even higher for what can be accomplished using large language models (LLMs). The family gets two new members: Gemini 1.5 Pro – An upgraded version of its February launch with better performance in all capabilities and benchmarks Gemini 1.5 Flash – A slimmed-down version tuned for efficiency with good results and few quality compromises. These models are particularly strong in long-context retrieval from a variety of modalities, including setting the state-of-the-art in tasks like long-document question answering, long-video analysis, and long-context automatic speech recognition. They have matched or exceeded the performance of Gemini 1.0 Ultra on a very broad set of evaluations.

Model Size and Architecture:

The Gemini 1.5 Pro is based on a sparse mixture-of-experts (MoE) Transformer architecture. Building upon the multimodal capabilities of Gemini 1.0, it harnesses vast MoE research at Google and in the broader AI field. MoE models route inputs to specific subsets of the model's parameters via a learned routing function. This can be thought of as a means of performing conditional computation, allowing for a much higher total parameter count with a constant number of active parameters per input. Key architectural improvements include long-context understanding enhancements, allowing Gemini 1.5 Pro to process up to 10 million tokens with no loss in performance. That means it could process almost five days of audio, more than ten hours of video, or the full text of "War and Peace" ten times over. And with that, making the model even more compute-efficient than its predecessors, with comparable results to Gemini 1.0 Ultra with a substantial reduction in training compute requirements.[5]

Training Infrastructure and Dataset:

Similar to the Gemini 1.0 series, the Gemini 1.5 models are trained using multiple 4096-chip pods of Google's TPUv4 accelerators, which are distributed across various data centers. The training utilizes a diverse range of multimodal and multilingual data. The pre-training dataset encompasses information from numerous domains, including web documents and code, and includes image, audio, and video content. During the instruction-tuning phase, the Gemini 1.5 models were fine-tuned on a dataset of multimodal information that pairs instructions with corresponding responses, along with additional tuning based on human preference data. For more detailed information, readers are encouraged to consult the Gemini 1.0 Technical Report.[5]

Capabilities:

Gemini 1.5 pushes the state-of-the-art in long-context capabilities, achieving near-perfect retrieval rates (over 99%) for up to 10 million tokens across text, video, and audio modalities. That is quite a big step up from models like Claude 3.0 (200k tokens) and GPT-4 Turbo with a maximum of 128k tokens.
Its practical applications have shown remarkable time savings, from 26% to 75%, in ten job categories.[5] Gemini 1.5 also exhibits emergent abilities, including learning to translate English into Kalamang, a language spoken by less than 200 people in the world, with proficiency comparable to a person trained using the same grammar book.[5]

Experimental Evaluations

Google tested Gemini 1.5's long-context abilities on both synthetic and real-world tasks.[5] On "needle-in-a-haystack"-style tasks, inspired by Kamradt (2023)[g], the model showed an exceptional ability in recalling information; it was able to identify important information within distracting contexts. Both Gemini 1.5 Pro and Flash achieved near-perfect (>99%) "needle" recall on text, video, and audio, even for context sizes up to several million tokens.[5] Tests demonstrate that when the context is expanded to 10 million tokens for all modalities, Gemini 1.5 Pro has sustained high recall performance, demonstrating the ability of handling unprecedented volumes of data while preserving accuracy.[5]

Figure reused from " Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context." by Team Gemini et al. (2024).

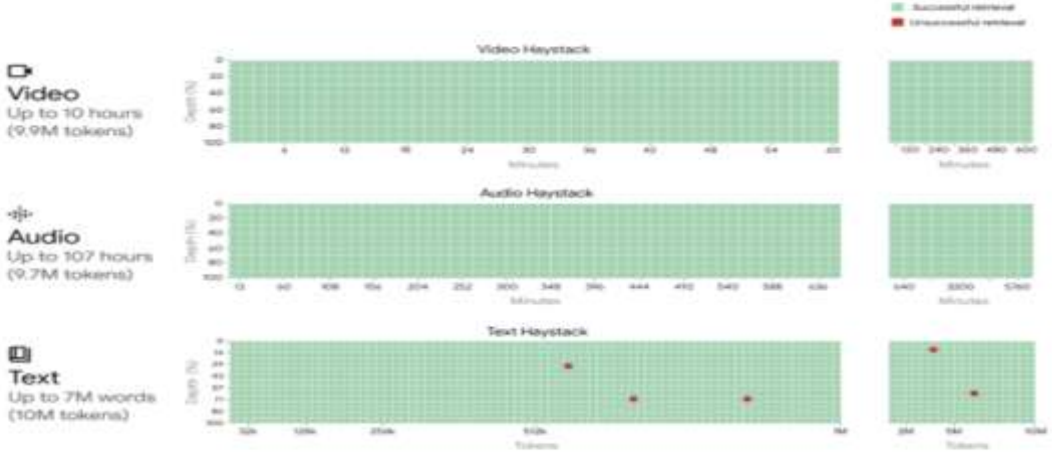


Figure 1.1 Gemini 1.5 Pro achieves near-perfect "needle" recall (>99.7%)

Key Insights

1. **Consistent Multimodal Retrieval**: Gemini 1.5 excels at retrieving relevant information in multimodal contexts, handling lengthy video, audio, and text inputs with high reliability.

2. **Exceptional Long-Context Handling**: For video and audio, the model demonstrates near-perfect recall (>99%) even with significantly extended contexts.
3. **Text Context Challenges**: While the model remains highly effective, some minor retrieval errors are observed at the highest token counts, indicating areas for further improvement in long-text processing.

Meta LlaMa 3.1

Llama 3 is a distinctive collection of language models developed to assist in multilingual processing, coding tasks, logical reasoning, and integration with the best possible means of external tools. Its flagship model is a dense Transformer with a staggering 405 billion parameters and a context window that can go up to a whopping 128,000 tokens.[6] Indeed, through extensive testing, Llama 3 has shown to perform competitively with state-of-the-art models like GPT-4 on a wide range of tasks. This release includes pre-trained and fine-tuned variants of the 405 billion parameter model, as well as a specialized version called Llama Guard 3, which focuses on ensuring the safety of both inputs and outputs.[6] Beyond its base capabilities, Llama 3 has been experimentally expanded to include image, video, and speech capabilities through a compositional approach. These extensions have achieved competitive results on state-of-the-art recognition tasks. While these models are still under development and not yet widely available.[6]

With Llama 3.1, Meta aims to provide the AI community with more speed and innovation using open-source tools, given its highly evolved architecture and extended training process, which can adapt to a variety of applications—something which will probably make this number one on the lists of many developers looking to harness the power of AI in their projects.

Architecture and Model Size

Llama 3.1: The dense transformer architecture succeeds its lineage—Llama 3—with 405 billion parameters. However, the really cool and astounding thing about this model is the extraordinary ability, sporting a context window of up to 128,000 tokens. It truly is the really big one in terms of knowledge base; the proficiently good and multi-lingual, for reasoning and logical tasks; the great on coders' issues; the grand at tools' integrations. Empirical evaluations show it to have a competitive advantage, often matching or even outperforming state-of-the-art models like GPT-4.

Moreover, it allows this architecture design to easily integrate with other multimodal inputs in a compositional way: images, videos, and speech. This versatility increases the possibility of applicability in many scenarios. Although all above multimodal capabilities are still under development, models—pre-trained and post-trained—are provided with safety mitigations through the Llama Guard 3 system. All told, Llama 3.1 is a tremendous advance in the capability of language model technology.[h]

Dataset Construction

The Llama 3.1 training data set is built to be as diverse as possible, containing multimodal, multilingual data in the form of web documents, code, images, audio, and video—making sure that a large number of domains are represented. Besides those measures, the dataset has gone through extensive cleaning processes by the company Meta in removing sensitive contents such as PII, adult material, and heavily duplicated contents in an attempt to present only high-quality datasets.[h][6]

Training data contains contributions up to the end of 2023 and is largely derived from web sources, which are parsed in a custom parser. Quality assurance involves heuristic and model-based filtering to include only high-quality tokens. The final distribution of the dataset is roughly 50% general knowledge, 25% mathematical and reasoning tasks, 17% code-related data, and 8% multilingual content. To this end, the benchmark performances were further improved by including annealing techniques during training.[h][6]

TRAINING METHODOLOGY

Llama 3 was trained on a dense Transformer architecture with 405 billion parameters, where the training has been designed with stability in mind, emphasizing efficiency by initializing the training using AdamW with a peak learning rate of 8×10^{-5}. Learning rates are annealed to a cosine schedule; the batch sizes start at 4 million tokens and increase to 16 million tokens as the model sees 2.87 trillion tokens during training.

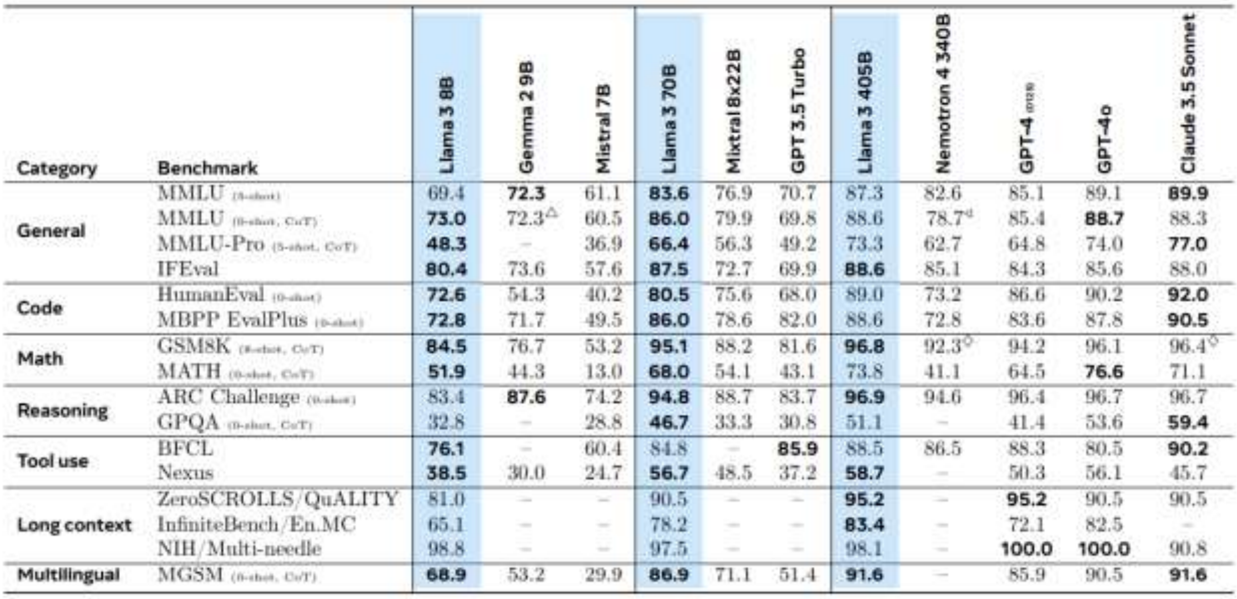
This gradual increase in batch size stabilizes training and prevents sudden loss spikes. Extensive empirical evaluations show this approach results in models that can deliver results comparable to other leading models in the industry, like GPT-4, across a wide array of tasks.[h]

Capabilities

The Llama 3 models are designed to perform well across a number of domains: multilingual text processing, logical reasoning, and tool-assisted tasks like coding. This is further enhanced by their ability to handle up to 128,000 tokens within a single context window for in-depth text analysis and generation. Moreover, Llama 3 adds support for image, video, and speech modalities through a compositional approach. It thus can be competitive in all recognition tasks belonging to each of these domains. Extensive testing confirms high-quality outputs of the model on a range of benchmarks.[6]

Another significant characteristic of the Llama 3 series is the Llama Guard 3 system, oriented toward safety in the monitoring of input and output interactions. These new developments ensure that Llama 3 is going to be a flexible and safe choice for a really large number of AI applications

Figure reused from "The llama 3 herd of models " by Dubey et al. (2024).



Category	Benchmark	Llama 3 8B	Gemma 2 9B	Mistral 7B	Llama 3 70B	Mixtral 8x22B	GPT 3.5 Turbo	Llama 3 405B	Nemotron 4 340B	GPT-4 (0125)	GPT-4o	Claude 3.5 Sonnet
General	MMLU (5-shot)	69.4	**72.3**	61.1	**83.6**	76.9	70.7	87.3	82.6	85.1	89.1	**89.9**
	MMLU (0-shot, CoT)	**73.0**	72.3△	60.5	**86.0**	79.9	69.8	88.6	78.7◁	85.4	**88.7**	88.3
	MMLU-Pro (5-shot, CoT)	**48.3**	–	36.9	**66.4**	56.3	49.2	73.3	62.7	64.8	74.0	**77.0**
	IFEval	**80.4**	73.6	57.6	**87.5**	72.7	69.9	**88.6**	85.1	84.3	85.6	88.0
Code	HumanEval (0-shot)	**72.6**	54.3	40.2	**80.5**	75.6	68.0	89.0	73.2	86.6	90.2	**92.0**
	MBPP EvalPlus (0-shot)	**72.8**	71.7	49.5	**86.0**	78.6	82.0	88.6	72.8	83.6	87.8	**90.5**
Math	GSM8K (8-shot, CoT)	**84.5**	76.7	53.2	**95.1**	88.2	81.6	**96.8**	92.3◇	94.2	96.1	96.4◇
	MATH (0-shot, CoT)	**51.9**	44.3	13.0	**68.0**	54.1	43.1	73.8	41.1	64.5	**76.6**	71.1
Reasoning	ARC Challenge (0-shot)	83.4	**87.6**	74.2	**94.8**	88.7	83.7	**96.9**	94.6	96.4	96.7	96.7
	GPQA (0-shot, CoT)	32.8	–	28.8	**46.7**	33.3	30.8	51.1	–	41.4	53.6	**59.4**
Tool use	BFCL	**76.1**	–	60.4	84.8	–	**85.9**	88.5	86.5	88.3	80.5	**90.2**
	Nexus	**38.5**	30.0	24.7	**56.7**	48.5	37.2	**58.7**	–	50.3	56.1	45.7
Long context	ZeroSCROLLS/QuALITY	81.0	–	–	90.5	–	–	**95.2**	–	**95.2**	90.5	90.5
	InfiniteBench/En.MC	65.1	–	–	78.2	–	–	**83.4**	–	72.1	82.5	–
	NIH/Multi-needle	98.8	–	–	97.5	–	–	98.1	–	**100.0**	**100.0**	90.8
Multilingual	MGSM (0-shot, CoT)	**68.9**	53.2	29.9	**86.9**	71.1	51.4	**91.6**	–	85.9	90.5	**91.6**

Table 2 Performance of finetuned Llama 3 models on key benchmark evaluations. The table compares the performance of the 8B, 70B, and 405B versions of Llama 3 with that of competing models. We **boldface** the best-performing model in each of three model-size equivalence classes. △Results obtained using 5-shot prompting (no CoT). ◁Results obtained without CoT. ◇Results obtained using zero-shot prompting.

This table shows how fine-tuning and scaling parameter sizes affect model performance on diverse NLP and reasoning tasks.

Takeaways

- **Llama 3 Models:** Scale effectively across benchmarks, particularly excelling in reasoning and multilingual tasks with the 405B model leading in many cases.
- **GPT-4:** Remains the strongest overall performer, especially in code, math, and reasoning tasks.
- **Claude:** Performs exceptionally in coding-related benchmarks.
- **Other Models (e.g., Mistral):** Fall behind in many categories but remain competitive in select benchmarks like reasoning.

Phi 2

We did not work on Phi 4 because, as of now, there isn't detailed public documentation on the exact architecture, dataset creation, or training methodology of Phi-4, Microsoft's most recent small language model. But from what is given, Phi-4 is a 14 billion-parameter model for increased mathematical reasoning and efficiency. The model reflects Microsoft's emphasis on high performance with smaller-scale architectures, balancing reasoning capability with computational efficiency.[j][k] Thus, we focus on the Microsoft Phi-2 model family, which is part of a line of small, efficient language models designed to achieve high performance on language tasks with reduced computational demands. It's an extensively optimized small LLM, aiming at a good performance on tasks involving reasoning, comprehension, and coding without trading off computational resource efficiency. This

2.7B-parameter model is innovative in "small-but-powerful" LLMs. Phi-2, developed on carefully curated datasets and knowledge transfer techniques, outperforms much larger models, like Llama-2-70B, in coding and reasoning tasks—showing that the quality of the data and scaling methods are competitive with the mere size of parameters.

Architecture and Model Size

Phi-2 falls under compact language models to approach or even surpass the performance of much larger models on the respective tasks. The detailed architectural specifics, such as the number of parameters, were not made publicly available; however, that is where the benefits of architecture for efficiency come into play to bring competitive results with smaller computational footprints. Phi-2 is a 2.7-billion parameter small language model (SLM) aimed at text generation and other language tasks. Even with a relatively small size, Phi-2 outperforms many bigger models, such as 7- and 13-billion parameter models, and scores close to the Llama-2 70B model.[l][m]

Training Compute and Dataset

While detailed information on the training compute for Phi-2 is not made available, it emphasizes effective architectures and optimization techniques to achieve high performance with a smaller model size. Phi-2 is optimized for efficiency and thus suitable for inference in resource-constrained environments, like laptops. It does especially well on systems with hardware optimized for AI workloads, including Intel Meteor Lake CPUs with Neural Processing Units (NPUs).[l][m] Phi-2 was trained on massive, diverse text datasets, including curated publicly available data. The focus was on pretraining with general-purpose text data and fine-tuning it for specific tasks, though precise datasets are not disclosed in detail.[l]

Training Method

It leverages state-of-the-art optimization techniques to minimize memory and computation overhead, including quantization. Quantization lowers the precision of weights and activations—for example, from fp16 to int8, or even 4-bit—which makes the model faster and lightweight during inference. The techniques related to quantization and pruning provide better latency and performance without the degradation of accuracy. Phi-2 is a language model based on a next-word prediction approach developed through a Transformer architecture. It was trained on 1.4 trillion tokens, in multiple passes over a combination of synthetic and web-based datasets, tuned for NLP and coding tasks. The training time took 14 days using 96 NVIDIA A100 GPUs.[l][m]

Unlike many other models, Phi-2 has not been aligned using techniques like reinforcement learning from human feedback (RLHF) or instruct fine-tuning. While that is so, it is, in several aspects related to toxicity and bias, better behaved than many other open source models that have undergone alignment processes as specific analyses, such as, for example, Figure 3. n. A subset of 6541 sentences are selected and scored from 0 to 1 based on scaled perplexity and sentence toxicity. The higher the score, the less likely it is that a model will produce toxic sentences instead of benign ones.[n]

Figure reused from " Phi-2: The Surprising Power of Small Language Models." by Team Microsoft et al. (2023).[n]

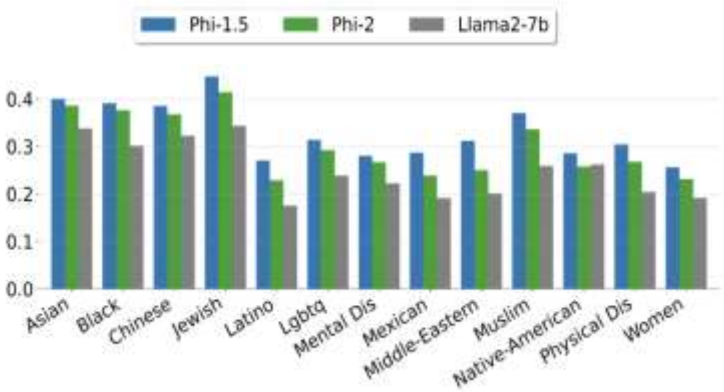


Figure 1.1 Safety scores computed on 13 demographics from ToxiGen.

Key Observations

1. **Overall Trends:**
   Phi-2 (green) generally shows lower bias scores compared to **Phi-1.5** (blue), suggesting improvements in reducing biases between iterations.

Llama2-7b (gray) often has lower or comparable bias scores, depending on the group, showing strong performance in mitigating biases.

2. **Group-Specific Observations:**
   Latino and Muslim groups: Phi-1.5 has notably higher bias scores compared to Phi-2 and Llama2-7b.
   LGBTQ, Mental Disabilities, and Physical Disabilities: All models show relatively lower scores, but Phi-1.5 is still slightly higher, indicating improvement in Phi-2 and Llama2-7b.
   Women and Native-American groups: Phi-2 and Llama2-7b exhibit lower bias, showing progress in addressing these sensitive categories.
3. **Phi-2 vs. Llama2-7b:**
   In most groups, **Phi-2** aligns closely with or performs better than **Llama2-7b**, highlighting its improvements in bias mitigation.

Takeaways

1. **Phi Model Improvements:** The shift from Phi-1.5 to Phi-2 demonstrates significant progress in reducing biases across multiple demographic groups.
2. **Llama2-7b Comparison:** Llama2-7b appears competitive or better in mitigating biases for most groups, reinforcing its role as a robust model for fairness.
3. **Remaining Challenges:** Certain groups (e.g., *Latino, Muslim*) still show higher bias scores, indicating areas where further refinement is needed.

PaLM 2

PaLM 2 improves over its predecessor, PaLM, in three major aspects: scaling, datasets, and architecture. The model is based on the "compute-optimal scaling" approach, where the size of the model and the size of the dataset have to be balanced well to achieve good performance. Among the most striking features of PaLM 2 has to be the much more diverse dataset, which is not limited to English but includes many other languages and even programming and mathematical domains. By this combination of pre-training goals, instead of just causal or masked language models, it enhances its knowledge of language in various contexts.[8]

Architecture and Model Size:

The Pathways Language Model (PaLM) is a Transformer architecture (Vaswani et al., 2017), but it is designed as a decoder-only model. This means that at every time step in the sequence, it can only attend to itself and previous steps, which makes it ideal for autoregressive tasks like text generation.
PaLM is a large model, with 540 billion parameters, trained on 780 billion tokens, which puts it among the largest models of its kind in the world. That's because it uses Google Pathways, which enables highly efficient training at scale—for large neural networks across thousands of TPU accelerator chips. This efficiency is one of the primary reasons PaLM can manage such enormous sizes without compromising on performance.[8][o]

\Training compute and Dataset:

The large PaLM 540B required a large setup: 6,144 TPU v4 chips spread across two TPU v4 Pods. These chips worked in concert with each other through a mix of model and data parallelism. This setup had a 57.8% hardware FLOPs utilization rate—the highest training efficiency recorded to date for large language models at this scale.[o]
The dataset for PaLM includes 780 billion tokens from a high-quality corpus, including filtered web pages, books, Wikipedia articles, news stories, open-source code repositories (like GitHub), and even social media conversations. Note that 50% of this massive dataset comes from social media. Hence, the model is really strong at conversing. Although PaLM has an exceptionally small amount of code in its dataset, it still outperforms by a large margin in coding tasks.[o]

METHODOLOGY

PaLM 2 significantly expands the predecessor by using a much larger dataset with much more diversity. It places an emphasis on non-English data to further improve its multilingual capabilities. The training included parallel data from hundreds of languages, which improves the model's ability in handling tasks associated with translation and multilingual settings. Special features like control tokens for toxicity filtering were added, allowing the model to maintain high performance while reducing harmful outputs. The architecture of PaLM 2 also supports an increased context length, allowing it to process longer inputs better and reason over them more effectively without any loss of precision.[8]

Pathways System: PaLM 2 is the first-of-its-kind use of the Google Pathways system. The largest TPU-based configuration ever trained, with 6,144 TPU v4 chips. Training used data parallelism over two TPU Pods and standard data and model parallelism within each Pod. This is a huge scale-up from previous models, such as: GLaM and LaMDA, which used a single TPU v3 Pod, Megatron-Turing NLG with pipeline parallelism over 2,240 A100 GPUs; Gopher with multiple TPU v3 Pods with up to 4,096 TPU v3 chips. PaLM's remarkable 57.8% hardware FLOPs utilization rate was made possible by efficient parallelism techniques combined with a redesigned Transformer block—this enables the parallel computation of the attention and feedforward layers, independently further optimized by TPU compiler optimizations.[p]

Stable LM 2

StableLM 2 1.6B is introduced as the first model in a new series of language models by Stable AI. The model weights for both versions are available on Hugging Face for public download and use. At the time of the technical report release, StableLM 2 1.6B held the position of the leading open model under 2B parameters by a considerable margin. Due to its compact size, throughput measurements on various edge devices are also provided. Additionally, several quantized checkpoints are open-sourced, with performance metrics comparing them to the original model.

Model Architecture and Size

Stable LM 2 1.6B is a 1.6 billion parameter decoder-only language model pre-trained on 2 trillion tokens of diverse multilingual and code datasets for two epochs. The Stable LM 2 1.6B model is trained on 64 Amazon P4d instances, equipped with 512 NVIDIA A100 (40GB HBM2) GPUs. The model size and ZeRO stage 1 distributed optimization eliminate the need for model sharding. The training configuration adjusts parameters such as micro-batch size, gradient accumulation steps, and activation checkpointing granularity to optimize speed. A batch size of 8,388,608 tokens is used, achieving approximately 170 TFLOPs/s per device, with 54.5% model FLOPs utilization. This hardware utilization can be increased to approximately 200 TFLOPs/s (64% MFU) by adjusting data parallelism and increasing gradient accumulation steps, although this would also result in a longer iteration time.

The model is a causal, decoder-only transformer architecture, similar to LLaMA, with several key differences. Notably:

Position Embeddings: Rotary Position Embeddings are applied to the first 25% of the head embedding dimensions to enhance throughput, following prior methods.

Normalization: LayerNorm is used with learned bias terms, replacing RMSNorm.

Biases: All bias terms are removed from the feed-forward networks and multi-head self-attention layers, except for those in the key, query, and value projections.

Data

The performance of the model is influenced by the design decisions made during pre-training, including the selection of data sources and sampling weights. The approach taken in this model is similar to previous work, where the majority of the training data is sourced from datasets used in other large language models (LLMs), such as RefinedWeb, subsets of the Pile, RedPajama, and the Stack. Additionally, the dataset is supplemented with OpenWebText, OpenWebMath, and parts of CulturaX. Upon reviewing randomly sampled documents from the mC4 subset of CulturaX, it was found that excessive HTML boilerplate was present, leading to the exclusion of this section, retaining only the OSCAR subset. The model also incorporates FanFics, a subset of 50,000 documents from fanfiction.net, which were selected based on their lowest perplexity scores.

In line with prior work, several raw datasets were transformed into structured formats for downstream tasks such as summarization, question-answering, and sentiment analysis, with additional instruction data from other sources to form Restruct-v1. All datasets used in training are open-source and commercially usable, with the majority hosted on the Hugging Face Hub. The Restruct-v1 dataset can be reproduced using the methods and templates provided by the original source.

Figure reused from " Stable lm 2 1.6 b technical report." by Bellagente et al. (2024).[9]

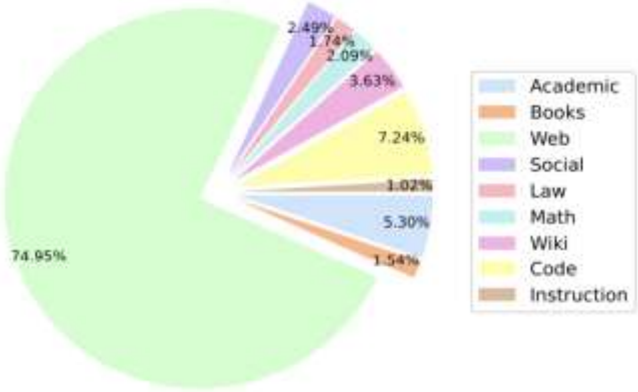


Figure 1.1 Percentage of effective training tokens by domain in the Stable LM 2 pre-training dataset.

Selecting the appropriate proportions of data from different domains, especially non-English and code data, is critical to achieving optimal performance. Multiple data mixes were tested and evaluated on downstream benchmarks to identify the most effective combination. Based on the results of these evaluations, the final training dataset was chosen, consisting of approximately 2 trillion tokens, with multilingual data in German, Spanish, French, Italian, Dutch, and Portuguese. The distribution of the dataset across different domains is visualized in the provided figure.

Training Methodology

StableLM 2 is trained to predict the next token using the standard autoregressive sequence modeling approach. The model is trained from scratch with a context length of 4096, utilizing FlashAttention-2's efficient sequence-wise parallelism optimizations. Training is conducted in BFloat16 mixed precision, while all-reduce operations are kept in FP32. To address instability from output logits divergence, a z-loss regularization term, zloss ∝ log2 Z, was added, though its impact on performance was minimal, leading to its exclusion from the final run. The model uses the AdamW optimizer with the following hyperparameters: β1 = 0.9, β2 = 0.95, ϵ = 1e-8, and λ (weight decay) = 0.1. Further details on the custom learning rate scheduler applied are available in Section 2.5.
A new learning rate scheduler is proposed, consisting of multiple stages to allow flexibility for continued pre-training. Initially, the learning rate is increased linearly to its maximum value of 1e−3 over 9,720 steps. This warm-up phase is followed by the main training phase, where the learning rate decreases according to the following formula:

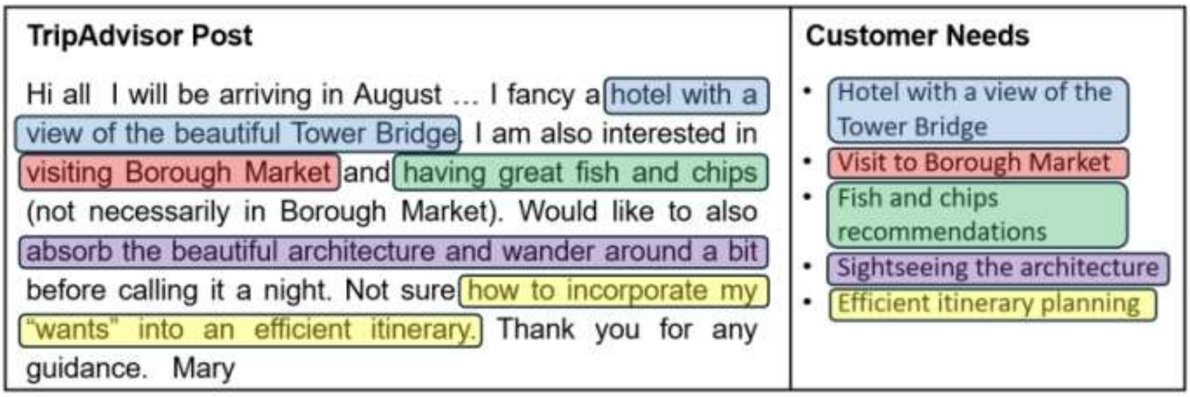
$$\begin{cases} m + (M - m) \times \frac{1}{2}\left[\cos\left(2\pi \times \frac{i}{N}\right) + 1\right] & \text{if } i \leq N/4 \quad \text{(cosine decay)} \\ \alpha \times \sqrt{i} + \beta & \text{if } i > N/4 \quad \text{(rsqrt decay)} \end{cases}$$

where m and M are the minimum and maximum learning rates, iii is the current step, and N is the total number of steps. The free parameters α and β are chosen to ensure the continuity of the scheduler and its derivative at i=N/4. The final stage of training involves linearly reducing the learning rate to zero over 80,000 steps, equating to around 670 billion tokens. The complete scheduler is illustrated in the provided figure, with additional details and ablation studies available in Appendix B.

Use cases of LLMs

The utilization of conversational agents powered by LLMs has the potential to reshape the customer needs management area, offering unparalleled efficiency and scalability. By harnessing the power of natural language understanding, these agents can interpret and respond to customer inquiries, identify solutions for new product development, and personalize recommendations.

Figure reused from “Automating Customer Needs Analysis: A Comparative Study of Large Language Models in the Travel Industry” by Barandoni et al. (2024).[16]



TripAdvisor Post

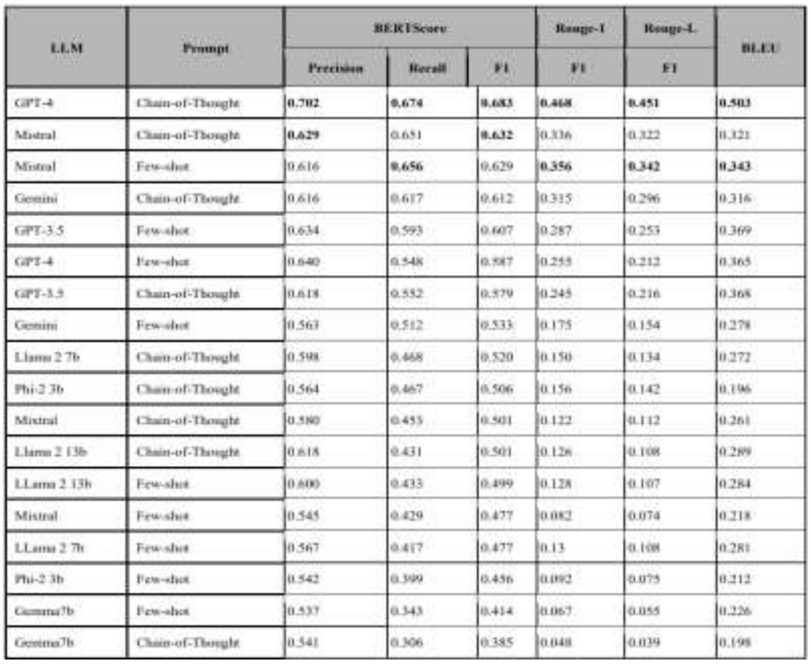
Hi all I will be arriving in August … I fancy a hotel with a view of the beautiful Tower Bridge. I am also interested in visiting Borough Market and having great fish and chips (not necessarily in Borough Market). Would like to also absorb the beautiful architecture and wander around a bit before calling it a night. Not sure how to incorporate my "wants" into an efficient itinerary. Thank you for any guidance. Mary

Customer Needs

- Hotel with a view of the Tower Bridge
- Visit to Borough Market
- Fish and chips recommendations
- Sightseeing the architecture
- Efficient itinerary planning

Example of a TripAdvisor post and related customer needs manually extracted through thematic analysis. Some of the needs, such as Hotel with a view of the Tower Bridge are explicit in the original text. Others, such as Efficient itinerary planning, are more implicit.

Figure reused from "Automating Customer Needs Analysis: A Comparative Study of Large Language Models in the Travel Industry" by Barandoni et al. (2024).[16]



LLM	Prompt	BERTScore			Rouge-1	Rouge-L	BLEU
		Precision	Recall	F1	F1	F1	
GPT-4	Chain-of-Thought	**0.702**	**0.674**	**0.683**	**0.468**	**0.451**	**0.503**
Mistral	Chain-of-Thought	**0.629**	0.651	**0.632**	0.336	0.322	0.321
Mistral	Few-shot	0.616	**0.656**	0.629	**0.356**	**0.342**	**0.343**
Gemini	Chain-of-Thought	0.616	0.617	0.612	0.315	0.296	0.316
GPT-3.5	Few-shot	0.634	0.593	0.607	0.287	0.253	0.369
GPT-4	Few-shot	0.640	0.548	0.587	0.255	0.212	0.365
GPT-3.5	Chain-of-Thought	0.618	0.552	0.579	0.245	0.216	0.368
Gemini	Few-shot	0.563	0.512	0.533	0.175	0.154	0.278
Llama 2 7b	Chain-of-Thought	0.598	0.468	0.520	0.150	0.134	0.272
Phi-2 3b	Chain-of-Thought	0.564	0.467	0.506	0.156	0.142	0.196
Mixtral	Chain-of-Thought	0.580	0.453	0.501	0.122	0.112	0.261
Llama 2 13b	Chain-of-Thought	0.618	0.431	0.501	0.126	0.108	0.289
LLama 2 13b	Few-shot	0.600	0.433	0.499	0.128	0.107	0.284
Mixtral	Few-shot	0.545	0.429	0.477	0.082	0.074	0.218
LLama 2 7b	Few-shot	0.567	0.417	0.477	0.13	0.108	0.281
Phi-2 3b	Few-shot	0.542	0.399	0.456	0.092	0.075	0.212
Gemma7b	Few-shot	0.537	0.343	0.414	0.067	0.055	0.226
Gemma7b	Chain-of-Thought	0.541	0.306	0.385	0.048	0.039	0.198

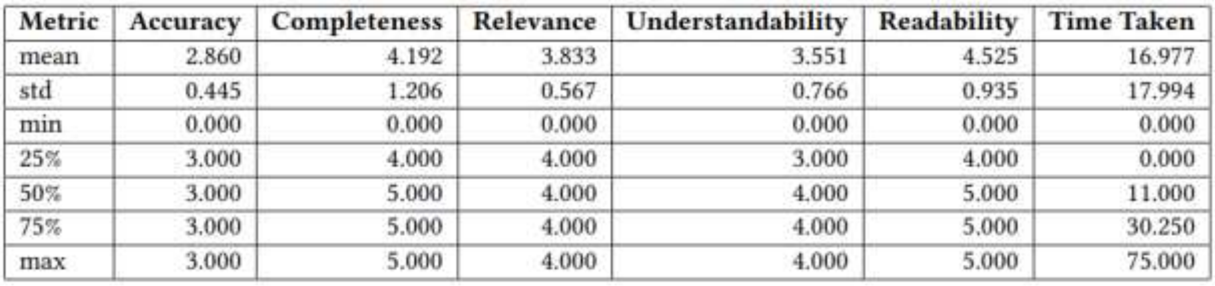
Results of the comparative analysis. For each LLM, the table presents the type of prompting employed and the scores obtained through the different metrics. The highest achieved values with proprietary and open-source models are highlighted in bold.

GPT-4, executed through optimized Chain-of-Thought prompting, has achieved the highest score in all the metrics employed. Mistral 7B is the second-best model, both with optimized CoT and non-optimized few-shot prompting: it outperformed all the other LLMs, Gemini and GPT-4 (standard few-shot) included. Despite being much smaller than the closed models assessed (GPT-4, GPT-3.5, and Gemini 1.0) in terms of numbers of parameters, it achieved a remarkably high quality of the responses. Surprisingly, GPT-3.5 achieved slightly better results through nonoptimized than through optimized prompting. Except for Mistral, all the open-source models obtained lower scores than the closed ones, especially through non optimized few-shot prompting. Gemma7b was the worst model in terms of accuracy.[16]

In terms of selecting the best model to perform travel customer analysis, we concluded that the best solution is Mistral 7B. It provided an extremely high quality of responses, the same of GPT-4, and its small size (seven billion parameters) allows it to be deployed on affordable machines. Furthermore, being open-source, it enables an elevated level of customization and security, as explained in the Introduction. Among the other open-source models, we observed satisfactory results also in Llama 2 7B and Phi-2 3B. With only three billion parameters, Phi-2 can be run on even smaller computer infrastructures. Therefore, it might be taken into consideration by practitioners and researchers with limited resources. Finally, we believe that it is possible to achieve much better scores with Llama 2 13B and Mixtral7Bx8: being larger than Llama 2 7B and Mistral, respectively, there is the possibility that their performance has been negatively influenced by our limited hardware.[16]

Large Language Models (LLMs) demonstrate significant superiority over humans in tasks related to code generation, particularly in producing documentation. Nearly all LLMs consistently surpass the quality of human-generated documentation. Our analysis further indicates that closed-source models, such as GPT-3.5, GPT-4, and Bard, outperform their open-source or source-available counterparts, such as LLaMA 2, across various performance metrics. Regarding efficiency, GPT-4 exhibits the slowest generation time, followed by LLaMA 2 and Bard, while ChatGPT displays comparable speeds. Notably, file-level documentation performs significantly worse across most parameters, excluding generation time, when compared to inline and function-level documentation. The following figures focus on GPT-3.5, GPT-4, Bard, LLama 2 and StarChat when evaluating each in terms of code generation.

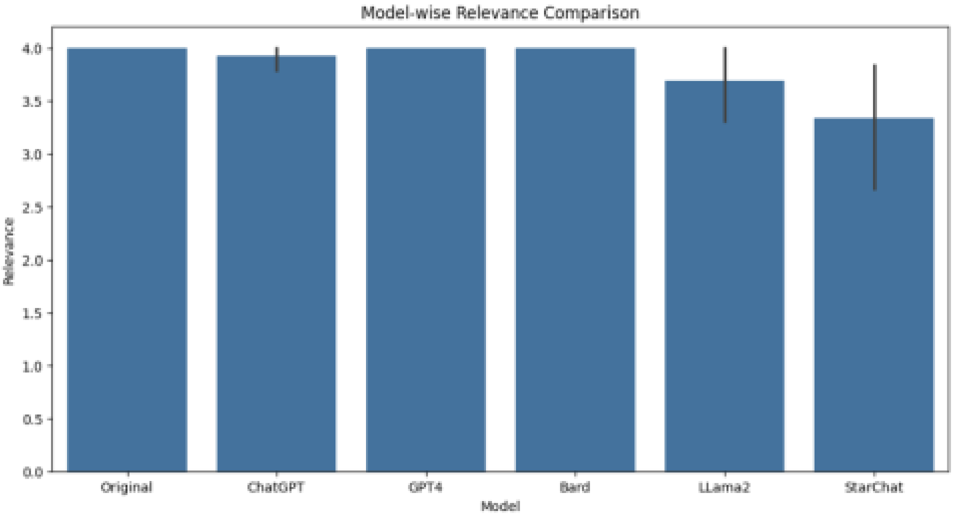
Figure reused from “A comparative analysis of large language models for code documentation generation” by Dvivedi et al. (2024).[13]



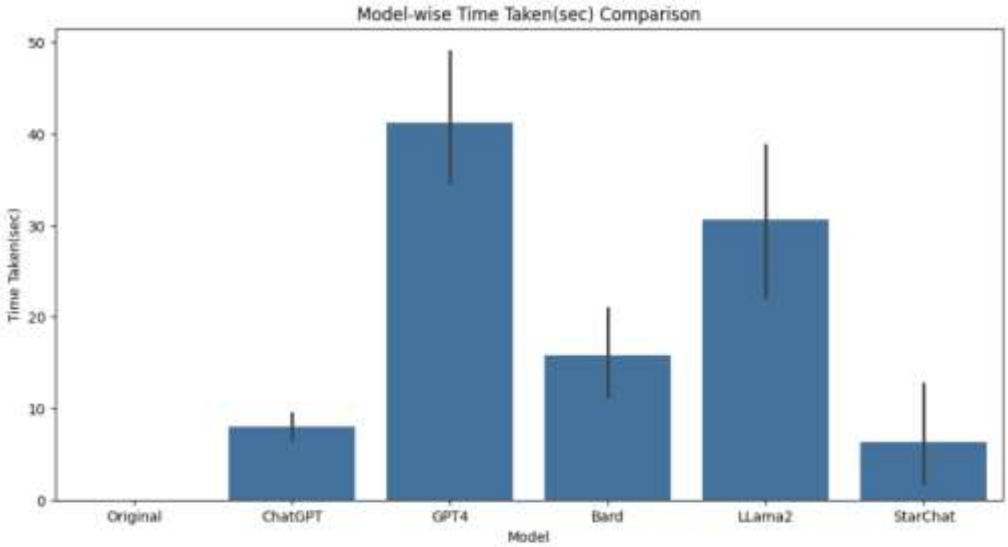
Metric	Accuracy	Completeness	Relevance	Understandability	Readability	Time Taken
mean	2.860	4.192	3.833	3.551	4.525	16.977
std	0.445	1.206	0.567	0.766	0.935	17.994
min	0.000	0.000	0.000	0.000	0.000	0.000
25%	3.000	4.000	4.000	3.000	4.000	0.000
50%	3.000	5.000	4.000	4.000	5.000	11.000
75%	3.000	5.000	4.000	4.000	5.000	30.250
max	3.000	5.000	4.000	4.000	5.000	75.000

Accuracy, Completeness, Relevance, Understandability, Readability and Time Taken averaged across all LLMs

Figure reused from “A comparative analysis of large language models for code documentation generation” by Dvivedi et al. (2024).[13]

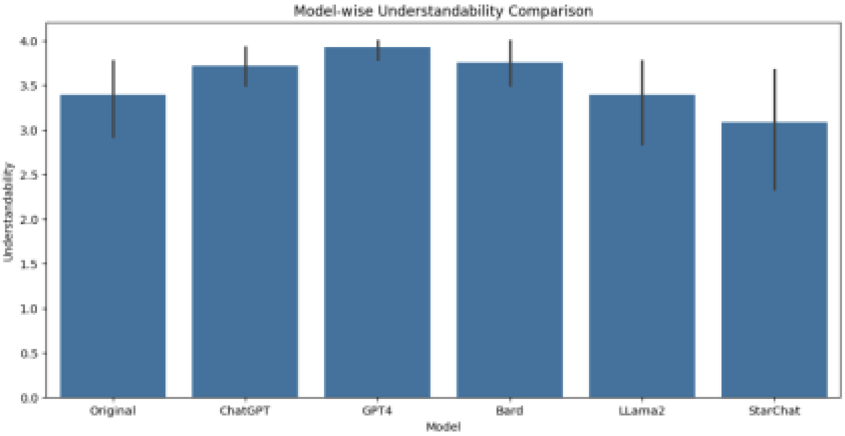


Model-wise relevance comparison
Figure reused from “A comparative analysis of large language models for code documentation generation” by Dvivedi et al. (2024).[13]



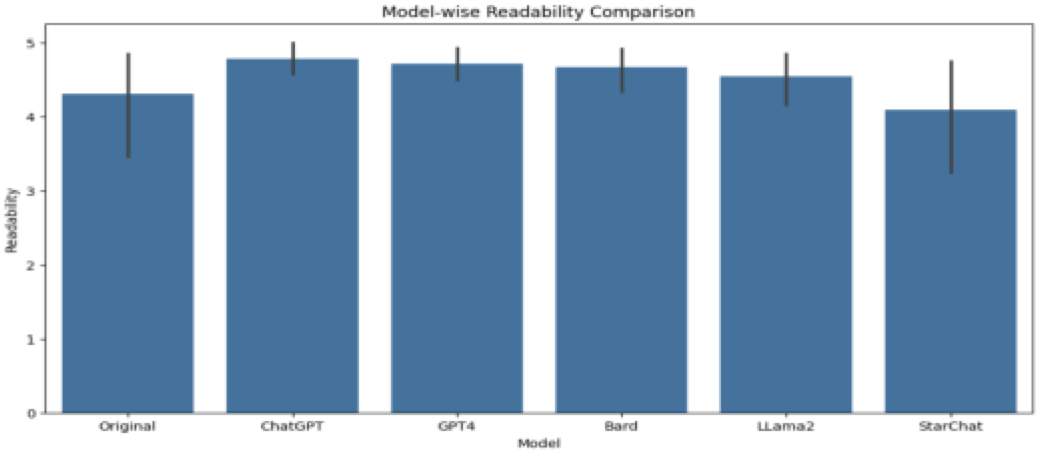
Model-wise Time taken comparison

Figure reused from "A comparative analysis of large language models for code documentation generation" by Dvivedi et al. (2024).[13]



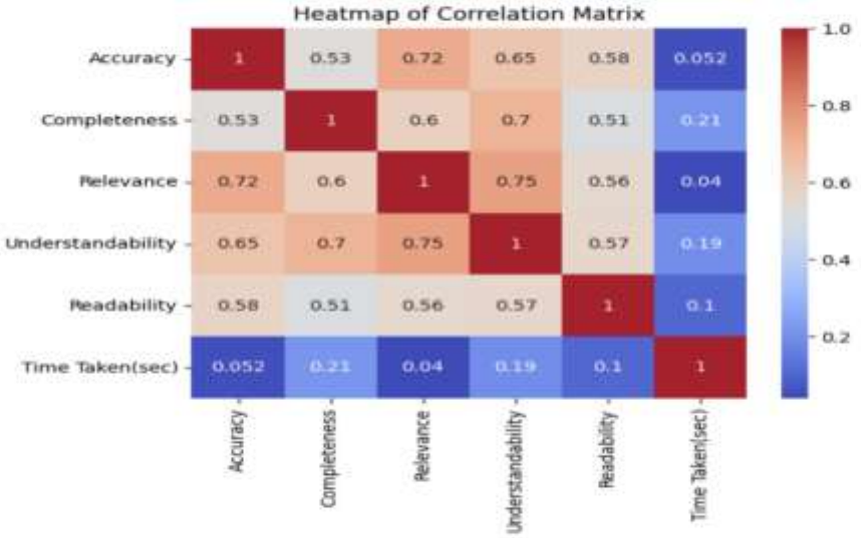
Model-wise understandability comparison

Figure reused from "A comparative analysis of large language models for code documentation generation" by Dvivedi et al. (2024).[13]



Model-wise readability comparison

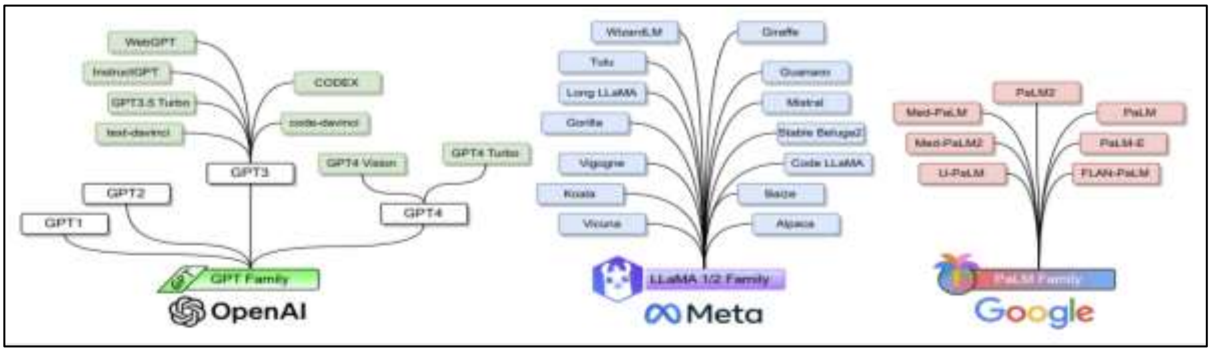
Figure reused from "A comparative analysis of large language models for code documentation generation" by Dvivedi et al. (2024).[13]



Metrics correlation heatmap

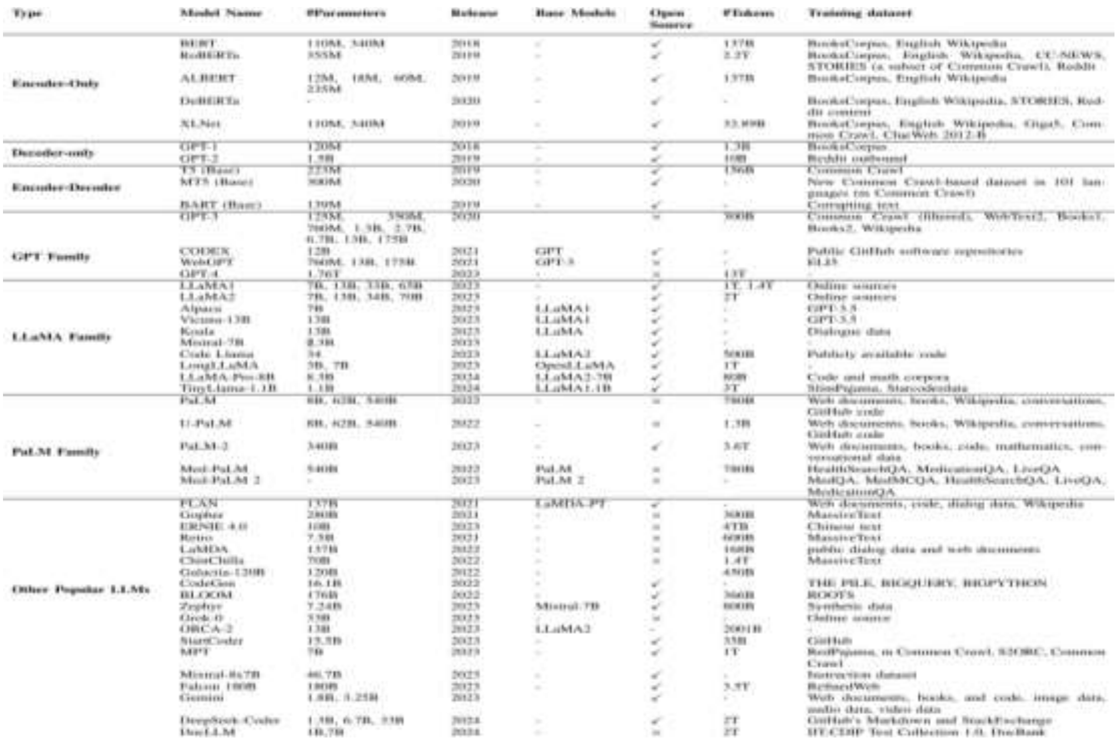
Comparative Evaluation

Figure reused from "Large Language Models: A Survey" by Minaee et al. (2024).[10]



Popular LLM Families

Figure reused from "Large Language Models: A Survey" by Minaee et al. (2024).[10]



Type	Model Name	#Parameters	Release	Base Models	Open Source	#Tokens	Training dataset
Encoder-Only	BERT	110M, 340M	2018	-	✓	137B	BooksCorpus, English Wikipedia
	RoBERTa	355M	2019	-	✓	2.2T	BooksCorpus, English Wikipedia, CC-NEWS, STORIES (a subset of Common Crawl), Reddit
	ALBERT	12M, 18M, 60M, 235M	2019	-	✓	137B	BooksCorpus, English Wikipedia
	DeBERTa	-	2020	-	✓	-	BooksCorpus, English Wikipedia, STORIES, Reddit content
	XLNet	110M, 340M	2019	-	✓	32.89B	BooksCorpus, English Wikipedia, Giga5, Common Crawl, ClueWeb 2012-B
Decoder-only	GPT-1	120M	2018	-	✓	1.3B	BooksCorpus
	GPT-2	1.5B	2019	-	✓	10B	Reddit outbound
Encoder-Decoder	T5 (Base)	223M	2019	-	✓	156B	Common Crawl
	MT5 (Base)	300M	2020	-	✓	-	New Common Crawl-based dataset in 101 languages (m Common Crawl)
	BART (Base)	139M	2019	-	✓	-	Corrupting text
GPT Family	GPT-3	125M, 350M, 760M, 1.3B, 2.7B, 6.7B, 13B, 175B	2020		×	300B	Common Crawl (filtered), WebText2, Books1, Books2, Wikipedia
	CODEX	12B	2021	GPT	✓	-	Public GitHub software repositories
	WebGPT	760M, 13B, 175B	2021	GPT-3	×	-	ELI5
	GPT-4	1.76T	2023	-	×	13T	-
LLaMA Family	LLaMA1	7B, 13B, 33B, 65B	2023	-	✓	1T, 1.4T	Online sources
	LLaMA2	7B, 13B, 34B, 70B	2023	-	✓	2T	Online sources
	Alpaca	7B	2023	LLaMA1	✓	-	GPT-3.5
	Vicuna-13B	13B	2023	LLaMA1	✓	-	GPT-3.5
	Koala	13B	2023	LLaMA	✓	-	Dialogue data
	Mistral-7B	7.3B	2023		✓	-	-
	Code Llama	34	2023	LLaMA2	✓	500B	Publicly available code
	LongLLaMA	3B, 7B	2023	OpenLLaMA	✓	1T	-
	LLaMA-Pro-8B	8.3B	2024	LLaMA2-7B	✓	80B	Code and math corpora
	TinyLlama-1.1B	1.1B	2024	LLaMA1.1B	✓	3T	SlimPajama, Starcoderdata
PaLM Family	PaLM	8B, 62B, 540B	2022	-	×	780B	Web documents, books, Wikipedia, conversations, GitHub code
	U-PaLM	8B, 62B, 540B	2022	-	×	1.3B	Web documents, books, Wikipedia, conversations, GitHub code
	PaLM-2	340B	2023	-	✓	3.6T	Web documents, books, code, mathematics, conversational data
	Med-PaLM	540B	2022	PaLM	×	780B	HealthSearchQA, MedicationQA, LiveQA
	Med-PaLM 2	-	2023	PaLM 2	×	-	MedQA, MedMCQA, HealthSearchQA, LiveQA, MedicationQA
Other Popular LLMs	FLAN	137B	2021	LaMDA-PT	✓	-	Web documents, code, dialog data, Wikipedia
	Gopher	280B	2021	-	×	300B	MassiveText
	ERNIE 4.0	10B	2023	-	×	4TB	Chinese text
	Retro	7.5B	2021	-	×	600B	MassiveText
	LaMDA	137B	2022	-	×	168B	public dialog data and web documents
	ChinChilla	70B	2022	-	×	1.4T	MassiveText
	Galactia-120B	120B	2022	-		450B	
	CodeGen	16.1B	2022	-	✓	-	THE PILE, BIGQUERY, BIGPYTHON
	BLOOM	176B	2022	-	✓	366B	ROOTS
	Zephyr	7.24B	2023	Mistral-7B	✓	800B	Synthetic data
	Grok-0	33B	2023	-	×	-	Online source
	ORCA-2	13B	2023	LLaMA2	-	2001B	-
	StartCoder	15.5B	2023	-	✓	35B	GitHub
	MPT	7B	2023	-	✓	1T	RedPajama, m Common Crawl, S2ORC, Common Crawl
	Mistral-8x7B	46.7B	2023	-	✓	-	Instruction dataset
	Falcon 180B	180B	2023	-	✓	3.5T	RefinedWeb
	Gemini	1.8B, 3.25B	2023		✓	-	Web documents, books, and code, image data, audio data, video data
	DeepSeek-Coder	1.3B, 6.7B, 33B	2024	-	✓	2T	GitHub's Markdown and StackExchange
	DocLLM	1B,7B	2024	-	×	2T	IIT-CDIP Test Collection 1.0, DocBank

Model	Parameters	Proprietary	Reference
GPT-4	1.8 trillion	OpenAI	Achiam et al., 2023
Claude 3	70 billion - 100 billion	Anthropic	Anthropic, 2024
Gemini 1.5	1.5 trillion	Google	Google, 2024
Meta Llama 3.1	405 billion	Meta	Meta, 2024
Phi-2	2.7 billion	Microsoft	Javaheripi et al., 2023
PaLM 2	PaLM 2	Google	Google, 2023
Stable LM 2	1.6 billion	Stability AI	Stability AI, 2024

Notes:

GPT-4: OpenAI's GPT-4 comprises of roughly 1.8 trillion parameters. More specifically, the architecture consists of eight models, with each internal model made up of 220 billion parameters.

Claude 3: Claude 3 models are large language models (LLMs) with parameter counts ranging from 70 billion (Claude 3 Haiku) to over 100 billion (Claude 3 Opus).

Gemini 1.5: With an incredible 1.5 trillion parameters, Gemini is one of the largest and most advanced language models developed to date.

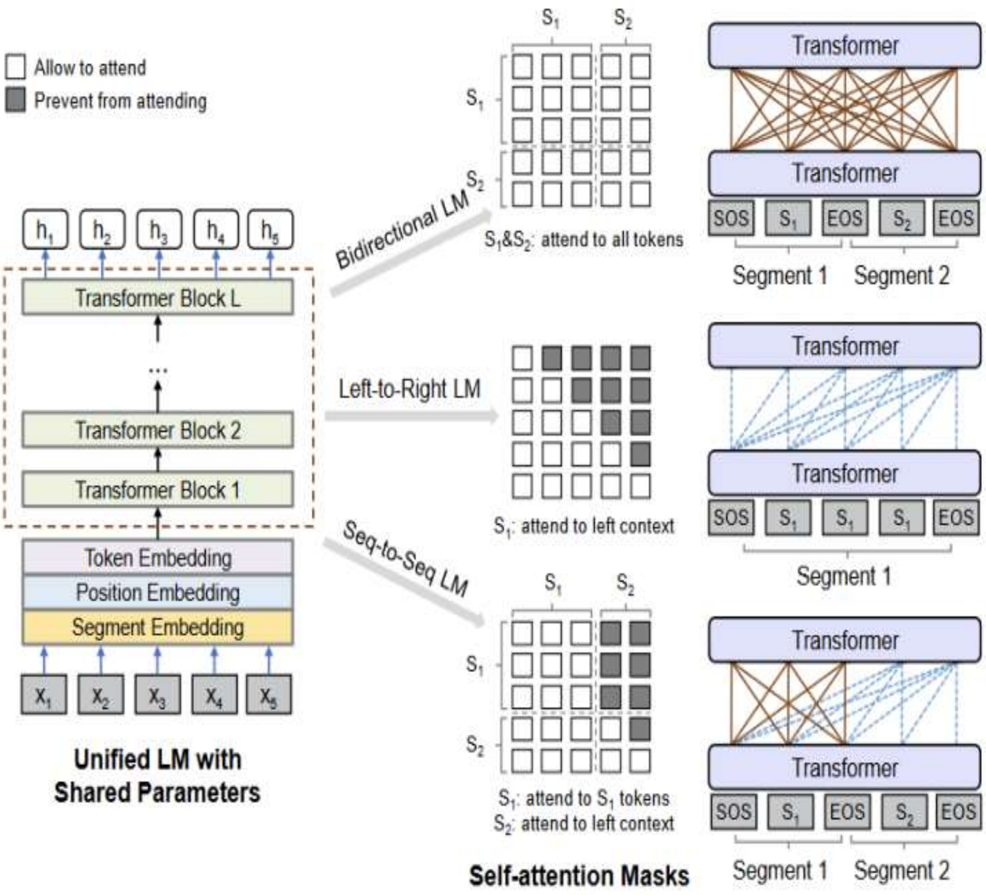
Meta Llama 3.1: Meta's Llama 3.1 is an open-source model with 405 billion parameters, released in 2024.

Phi-2: Microsoft's Phi-2 is an open-source model with 2.7 billion parameters, introduced in 2023.

PaLM 2: Google has not publicly disclosed the number of parameters for PaLM 2.

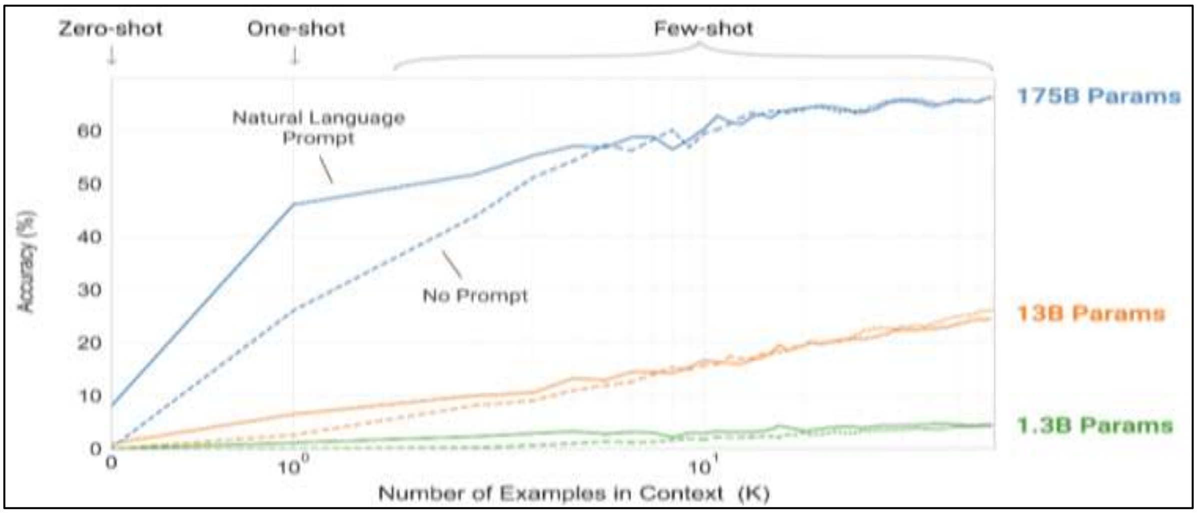
Stable LM 2: Information on Stable LM 2 is limited, and specific details about its parameters are not readily available.

Figure reused from "Electra: Pretraining text encoders as discriminators rather than generators" by K.Clark et al. (2020).[10]



Overview of unified LM pre-training. The model parameters are shared across the LM objectives (i.e., bidirectional LM, unidirectional LM, and sequence-to-sequence LM).

Figure reused from “Large Language Models: A Survey” by Minaee et al. (2024).[10]



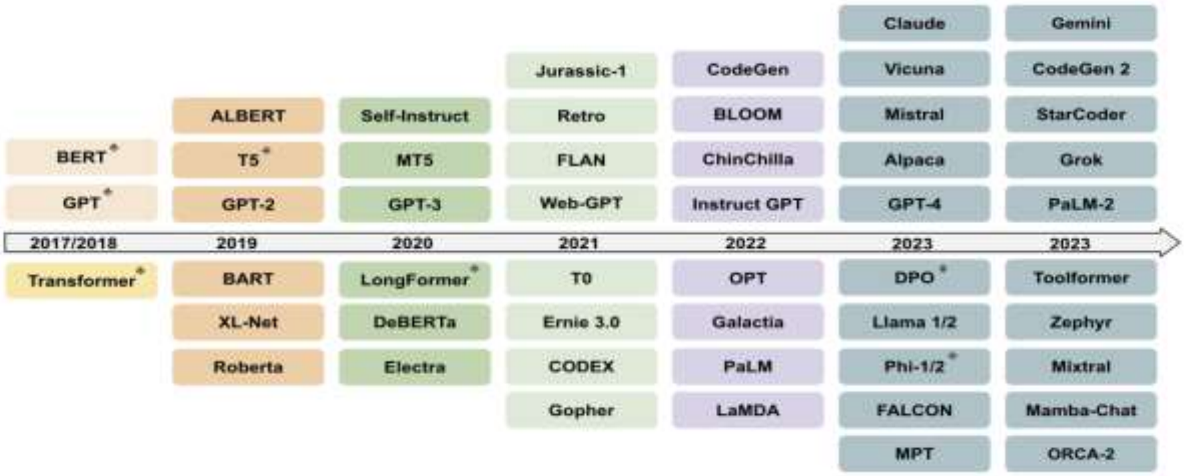
It can be observed that larger models make increasingly efficient use of in-context information. It shows in-context learning performance on a simple task requiring the model to remove random symbols from a word, both with and without a natural language task description.

Significance of Transformers in LLMs

Transformer: in a ground-breaking work [12], Vaswani et al. proposed the Transformer framework, which was originally designed for effective parallel computing using GPUs. The heart of Transformer is the (self-)attention mechanism, which can capture long-term contextual information much more effectively using GPUs than the recurrence and convolution mechanisms.

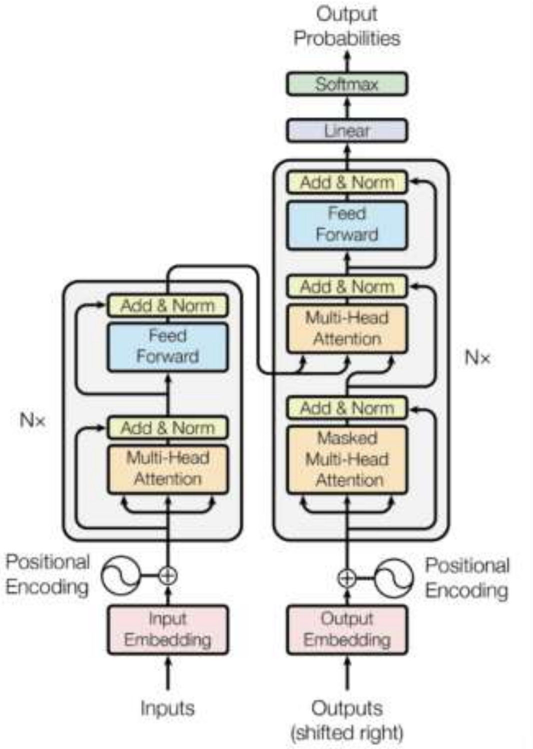
The Transformer language model architecture, originally proposed for machine translation, consists of an encoder and a decoder. The encoder is composed of a stack of N = 6 identical Transformer layers. Each layer has two sub-layers. The first one is a multi-head self-attention layer, and the other one is a simple position-wise fully connected feed-forward network. The decoder is composed of a stack of 6 identical layers. In addition to the two sub-layers in each encoder layer, the decoder has a third sub-layer, which performs multi-head attention over the output of the encoder stack. The attention function can be described as mapping a query and a set of keyvalue pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. Instead of performing a single attention function with dmodel dimensional keys, values and queries, it is found to be beneficial to linearly project the queries, keys and values h with different, learned linear projections to dk, dk and dv dimensions, respectively. Positional encoding is incorporated to fuse information about the relative or absolute position of the tokens in the sequence.

Figure reused from “Large Language Models: A Survey” by Minaee et al. (2024).[10]



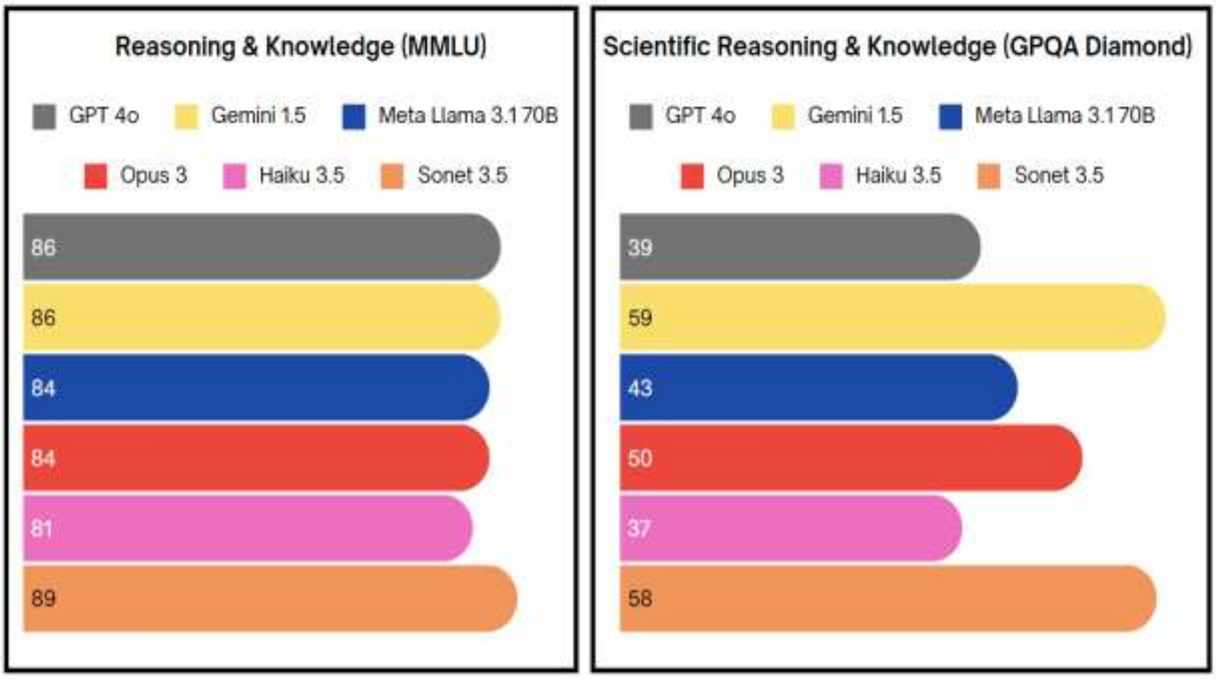
Timeline of some of the most representative LLM frameworks (so far). In addition to large language models with our #parameters threshold. A few representative works which pushed the limits of language models, and paved the way for their success (e.g. vanilla Transformer, BERT, GPT-1), as well as some small language models are also included.

Figure reused from "Large Language Models: A Survey" by Vaswani et al. (2017).[12]



The diagram presents the architecture of the Transformer model, a fundamental framework in natural language processing (NLP). The model takes in inputs (left) and produces outputs step by step (right), which makes it highly effective in tasks such as machine translation, text generation, and summarization. This design was introduced in the paper "Attention Is All You Need" by Vaswani et al. (2017).

Comparative Quality Evaluations:



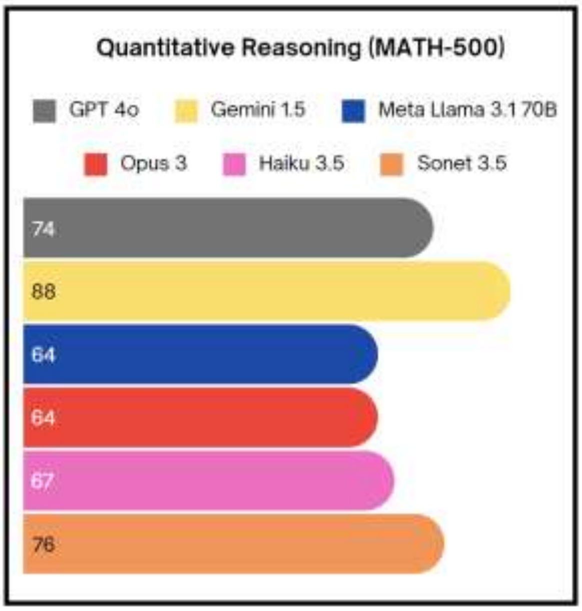
Reasoning & Knowledge (MMLU)

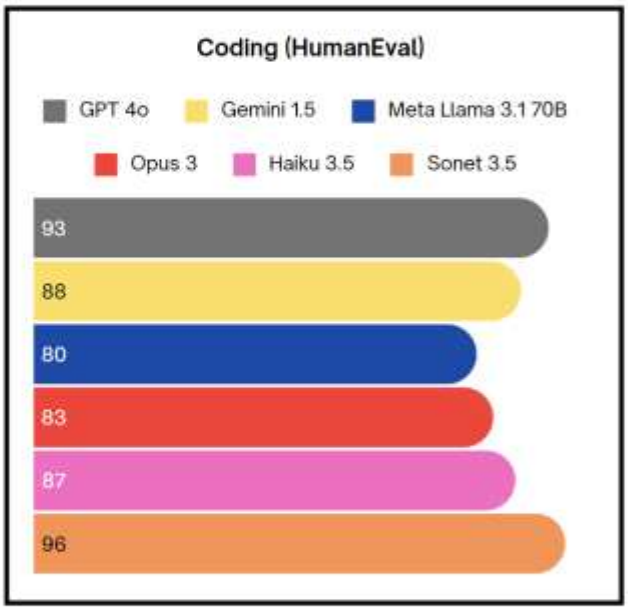
The MMLU benchmark is used to evaluate the reasoning and knowledge of various models in a variety of tasks. Out of the other models, Sonet3.5 appears to be the most proficient one - as it manages to acquire a score of 89 reflecting its general reasoning due to its sophisticated architecture or its large and diversly compiled dataset. Sitting right behind Sonet, is GPT-4 along with Gemini 1.5 who both share a notable score of 86 which may imply their similarity in terms of general knowledge and reasoning. Both Meta LLaMA 3.1 (70B) and Opus 3 managed to score 84 which although is a fair score but is still a bit below the top scorers showcasing a range of dropped features. As for Haiku 3.5, it managed to score 81 which is lacking greatly and may stem from an issue with the optimization method or the training data that was being used.

Scientific Reasoning & Knowledge (GPQA Diamond)

The GPQA Diamond benchmark evaluates how well the model is able to perform specialized scientific reasoning tasks. In this benchmark, the top contender appears to be Gemini 1.5 as it scored a solid 59 indicating that it might have been fine-tuned or trained further on scientific datasets. Starting Sonet 3.5, which scored 58, marking a further advancement in areas of specialized reasoning. Opus 3 earned a mark of 50 which is decent yet mid-tier as per the other models in this field. Meta LLaMA 3.1 70B also did not do stellar in the field and managed to score 43 showcasing only a decent understanding of the scientific tasks.

The results indicate notable differences between the models in general and specialized performance. **Sonet 3.5** and **Gemini 1.5** consistently perform well across both benchmarks, showcasing their versatility. In contrast, **GPT-4**, despite its strong general reasoning abilities, demonstrates a relative weakness in scientific reasoning, possibly due to a lack of emphasis on domain-specific fine-tuning. **Haiku 3.5**, while scoring consistently lower in both benchmarks, still performs respectably, suggesting potential improvements with further training or optimization.





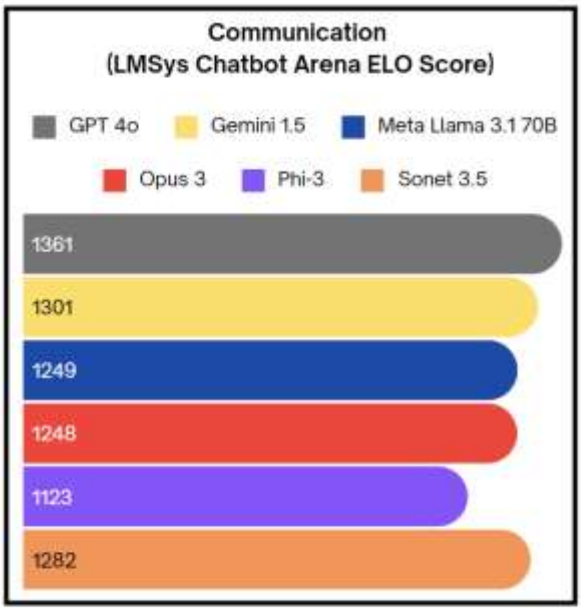
Quantitative Reasoning (MATH-500)

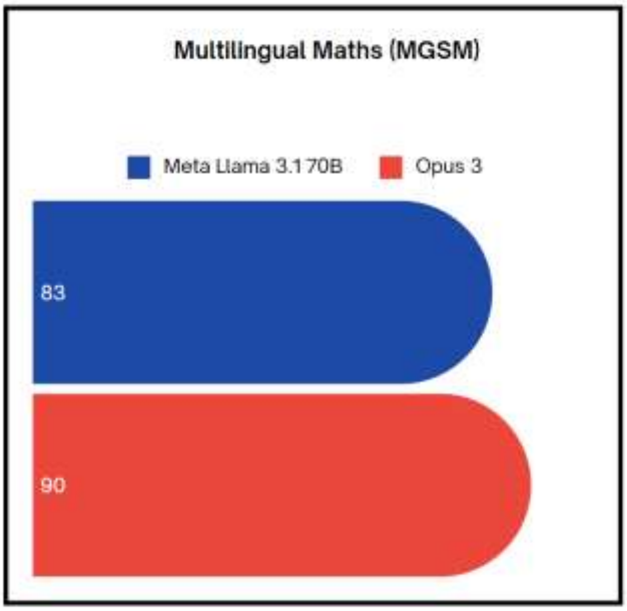
The MATH-500 benchmark checks how well models handle tough math tasks, including quantitative reasoning. Gemini 1.5 tops the list with a score of 88, indicating its superiority in numerical and symbolic reasoning. This result highlights Gemini's capability, mostly acquired from extensive training on mathematical datasets. Sonet 3.5 follows with a score of 76, showcasing strong quantitative reasoning skills, still not at the level of Gemini 1.5 though. GPT-4, a widely recognized leader in general AI capabilities, scores 74 - putting it third here, which means it's solid but not amazing in specialized quantitative tasks. Meta LLaMA 3.1 (70B) and Opus 3 both get scores of 64, reflecting adequate mathematical abilities, potentially limited by their training focus. maybe held back by their training focus. Haiku 3.5 earns a score of 67 – it beats both Meta LLaMA and Opus by showing fair skill in quantitative tasks.

Coding (HumanEval)

The HumanEval benchmark is used to evaluate the models' ability to generate functional code for programming tasks. Sonnet 3.5 tops with an astonishing 96 score, showing extraordinary coding capability and adaptability to any kind of programming challenges. GPT-4 follows closely with a score of 93, which again suggests strong coding capabilities and great versatility. Gemini 1.5 scores a respectable 88, but still below Sonet or GPT-4, so there is still a little room to go on programming-centric tasks. Haiku 3.5 scores 87, indicating strong performance—likely helped by seeing coding-specific datasets. Opus 3 shows very respectable coding abilities with a score of 83. Meta LLaMA 3.1 (70B) scores 80, the lowest score on this benchmark but still a very respectable level of fundamental coding ability.

Results show large differences in performance between models across the MATH-500 and HumanEval benchmarks. Gemini 1.5 performs very well on quantitative reasoning but lags slightly in coding, which definitely means it focuses on mathematical datasets. In contrast, while Sonet 3.5 is a strong leader in tasks related to coding, it has relatively lower performance on quantitative reasoning, hence optimized for programming-related tasks. GPT-4 is relatively balanced on both benchmarks but does not take the top on either, reflecting its general-purpose nature rather than specialization. This comparative analysis underlines the importance of task-specific fine-tuning in LLMs. For coding and software development applications, Sonet 3.5 and GPT-4 are the top choices. On mathematical reasoning tasks, Gemini 1.5 is unparalleled. This analysis shows that no model dominates all benchmarks, so a model selection should be performed on the basis of the specific application domain. Further developments in LLMs must be focused on how to balance these abilities for wider applicability.





Communication (LMSys Chatbot Arena ELO Score)

The LMSys Chatbot Arena ELO Score is a measure of the communicative and conversational abilities of large language models, LLMs. In this benchmark, GPT-4 shines with 1361 points, as it can really hold quite coherent and context-aware conversations. This score indicates that GPT-4 is currently the most fine-tuned conversational agent among the ones tested. In second place is Gemini 1.5, which scores an astonishing 1301 in this benchmark, showing how much it has advanced on the state-of-the-art dialogue generation tasks. Sonet 3.5 comes on at 1282, indicating strong conversational capabilities but still a bit behind Gemini.

Meta LLaMA 3.1 (70B) and Opus 3 are closely matched, with scores of 1249 and 1248, respectively, indicating competitive but slightly less polished conversational abilities. Phi-3, at 1123, ranks the lowest in this benchmark, with room for improvement in natural language understanding and conversational coherence. More generally, this chart singles out GPT-4 as dominant in communication; the other models show quite a range of competence.

Multilingual Maths (MGSM)

The Multilingual Generalist Mathematics (MGSM) benchmark tests mathematical problem-solving capabilities of models in many languages. Under this category, Opus 3 takes the lead with a score of 90, which shows the model's great ability to handle math reasoning tasks for different languages. It showcases the strong optimization of multilingual and mathematical datasets by Opus 3. Meta LLaMA 3.1 (70B) scores 83, showing strong math reasoning but still behind Opus 3.

The results show a large performance gap, where Opus 3 is much better at multilingual mathematical tasks, probably due to some special optimizations during training. Meta LLaMA 3.1 is strong but less specialized for this task. GPT-4 is unmatched in conversational abilities and sets the bar high for natural language understanding in LLMs. On the other hand, Opus 3 has unparalleled expertise in multilingual mathematical reasoning, suggesting a more domain-specific focus. Gemini 1.5 and Sonet 3.5 demonstrate balanced performance in communication, indicating versatility but not domain dominance.

Cross-Benchmark Observations

1. **GPT-4** consistently tops the list in communication and general reasoning tasks, setting a high watermark for conversational AI. However, it lags behind other state-of-the-art models in scientific reasoning and quantitative reasoning.
2. **Gemini 1.5** also performs strongly on the reasoning and quantitative tasks and is a good all-rounder, being especially strong in scientific reasoning and mathematics.
3. **Meta LLaMA 3.1 (70B)** looks quite competitive in all categories, never hitting high performance anywhere, speaking of general competence with room for domain-specific optimization.
4. **Opus 3** is specialized to multilingual mathematical reasoning, thus intense on diverse languages and quantitative tasks.
5. **Sonet 3.5** has great coding and general reasoning but mediocre results everywhere else.
6. **Haiku 3.5** is steady in mid-tier performance but does not lead in any benchmark, which could suggest it is less optimized for high-complexity tasks.
7. **Phi-3** struggles across the board, especially in communication, indicating that it is even less refined than its competitors.

These results demonstrate the diversification of the capabilities of LLMs and some of the model design trade-offs. GPT-4 is a clear winner on conversational and general reasoning tasks, while models such as Gemini 1.5 and Opus 3 outperform on harder specific tasks, for example, scientific reasoning and multilingual mathematics. No single model has universally excelled all benchmarks; analysis emphasizes the need to drive model selection based on task-specific requirements.

Note:

The results reported in this paper are based on benchmark tests made publicly available through the Artificial Analysis AI website[q]. Although these benchmarks do provide incredibly useful insight into the relative performance of state-of-the-art large language models on a wide range of tasks, it should be noted that the source does come with the following warning: The website explicitly states:

"Our goal is to provide high-quality benchmarks, useful in decision-making; however, please note that we give no guarantees or warranties, and performance may vary for particular use cases. Performance can be different depending on the particular needs, so we recommend further evaluation for the best fit."

As such, the reported performance metrics, while indicative of the capabilities of the models under standardized conditions, do not necessarily fully capture their behavior in specific real-world applications. This therefore calls for further evaluations on specific use cases to be carried out by researchers and practitioners alike before selecting a model. It will ensure the LLM selected will be well aligned with the needs of the task or domain of interest.

REFERENCES

[1] Achiam, Josh, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida et al. "Gpt-4 technical report." *arXiv preprint arXiv:2303.08774* (2023).

[3] Nori, Harsha, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. "Capabilities of gpt-4 on medical challenge problems." *arXiv preprint arXiv:2303.13375* (2023).

[4] Agrawal, Pravesh, Szymon Antoniak, Thomas Wang, and others. "The Claude 3 Model Family: Opus, Sonnet, Haiku." *In Proceedings of the Conference on Artificial Intelligence*, 2023. https://api. semanticscholar.org/CorpusID:268232499.

[5] Team, Gemini, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer et al. "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context." *arXiv preprint arXiv:2403.05530* (2024).

[6] Dubey, Abhimanyu, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur et al. "The llama 3 herd of models." *arXiv preprint arXiv:2407.21783* (2024).

[7] Adler, Bo, Niket Agarwal, Ashwath Aithal, Dong H. Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper et al. "Nemotron-4 340B Technical Report." *arXiv preprint arXiv:2406.11704* (2024).

[8] Anil, Rohan, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri et al. "Palm 2 technical report." *arXiv preprint arXiv:2305.10403* (2023).

[9] Bellagente, Marco, Jonathan Tow, Dakota Mahan, Duy Phung, Maksym Zhuravinskyi, Reshinth Adithyan, James Baicoianu et al. "Stable lm 2 1.6 b technical report." *arXiv preprint arXiv:2402.17834* (2024).

[10] Minaee, Shervin, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. "Large language models: A survey." *arXiv preprint arXiv:2402.06196* (2024).

[11] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pretraining text encoders as discriminators rather than generators," arXiv preprint arXiv:2003.10555, 2020.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.

[13] Dvivedi, Shubhang Shekhar, Vyshnav Vijay, Sai Leela Rahul Pujari, Shoumik Lodh, and Dhruv Kumar. "A comparative analysis of large language models for code documentation generation." In *Proceedings of the 1st ACM International Conference on AI-Powered Software*, pp. 65-73. 2024.

[14] Junaed Younus Khan and Gias Uddin. 2022. Automatic Code Documentation Generation Using GPT-3. In Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering. ACM, Rochester MI USA, 1–6. https://doi.org/10.1145/3551349.3559548

[15] Juho Leinonen, Paul Denny, Stephen MacNeil, Sami Sarsa, Seth Bernstein, Joanne Kim, Andrew Tran, and Arto Hellas. 2023. Comparing Code Explanations Created by Students and Large Language Models. In Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1 (, Turku, Finland,) (ITiCSE 2023). Association for Computing Machinery, New York, NY, USA, 124–130. https: //doi.org/10.1145/3587102.3588785

[16] Barandoni, Simone, Filippo Chiarello, Lorenzo Cascone, Emiliano Marrale, and Salvatore Puccio. "Automating Customer Needs Analysis: A Comparative Study of Large Language Models in the Travel Industry." *arXiv preprint arXiv:2404.17975* (2024).

[17] Vlačić, B., Corbo, L., e Silva, S. C., & Dabić, M. (2021). The evolving role of artificial intelligence in marketing: A review and research agenda. Journal of Business Research, 128, 187-203. https://doi.org/10.1016/j.jbusres.2021.01.055.

[18] Zhou, F., Ayoub, J., Xu, Q., & Jessie Yang, X. (2020). A machine learning approach to customer needs analysis for product ecosystems. Journal of mechanical design, 142(1), 011101. https://doi.org/10.1115/1.4044435.

[19] Soni, V. (2023). Large language models for enhancing customer lifecycle management. Journal of Empirical Social Science Studies, 7(1), 67-89.

[20] Wu, F. (2020). BERTScore Default Layer Performance on WMT16. Retrieved from: https://docs.google.com/spreadsheets/d/1RKOVpselB98Nnh_EOC4A2BYn8_201tmPODpNWu4w7xI/edit?usp=shar ing. Accessed March 2024.

Links

[a] Stanford Center for Research on Foundation Models. "OpenAI GPT-4 Report." *CRFM Foundation Model Transparency Index*. Accessed December 9, 2024. https://crfm.stanford.edu/fmti/May-2024/company-reports/OpenAI_GPT-4.html.

[b] "GPT-4." *Wikipedia*. Last modified November 28, 2024. Accessed December 9, 2024. https://en.wikipedia.org/wiki/GPT-4.

[c] "OpenAI. *GPT-4 Technical Report*." 2023. Accessed December 9, 2024. https://openai.com/research/gpt-4.

[d] "Introducing the next generation of Claude." Accessed December 9, 2024. https://www.anthropic.com/news/claude-3-family?form=MG0AV3

[e] "Claude 3 Haiku: our fastest model yet." Accessed December 9, 2024. https://www.anthropic.com/ news/claude-3-haiku

[f] "Claude 3.5 Sonnet." Accessed December 9, 2024. https://www.anthropic.com/claude/sonnet? form=MG0AV3

[g] Gregory Kamradt, 2023. Accessed December 9, 2024. URL https://github.com/gkamradt/LLMTest_ NeedleInAHaystack/blob/main/README.md.

[h] "Introducing Llama 3.1: Our most capable models to date." Published July 23, 2024. Accessed December 9, 2024. https://ai.meta.com/blog/meta-llama-3-1/

[i] "A 2024 Outlook for Large Language Models (LLMs) " Published December 20, 2023. Accessed December 20, 2024. https://www.evolvedash.com/blog/llms-in-2024/

[j] Microsoft Tech Community. "Introducing Phi-4: Microsoft's Newest Small Language Model Specializing in Complex Reasoning." Last modified December 2024. Accessed December 20, 2024 .https:// techcommunity.microsoft.com/blog/aiplatformblog/introducing-phi-4-microsoft's-newest-small-language-model-specializing-in-comple/4357090.

[k] MarkTechPost. "Microsoft AI Introduces Phi-4: A New 14-Billion-Parameter Small Language Model Specializing in Complex Reasoning." Last modified December 12, 2024. Accessed December 20, 2024.https://www.marktechpost.com/2024/12/12/microsoft-ai-introduces-phi-4-a-new-14-billion-parameter-small-language-model-specializing-in-complex-reasoning/.

[l] Hugging Face. "A Chatbot on your Laptop: Phi-2 on Intel Meteor Lake." Last modified December 2024. Accessed December 20, 2024. https://huggingface.co/blog/phi2-intel-meteor-lake.

[m] Hugging Face. "PhineTuning 2.0." Last modified December 2024. Accessed December 20, 2024. https://huggingface.co/blog/g-ronimo/phinetuning.

[n] Microsoft Research. "Phi-2: The Surprising Power of Small Language Models." Last modified June 22, 2023. Accessed December 20, 2024. https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/.

[o] "PaLM." *Wikipedia*. Last modified December 18, 2024. Accessed December 20, 2024. https://en.wikipedia.org/wiki/PaLM.

[p] Google Research. "Pathways Language Model (PaLM): Scaling to 540 Billion Parameters for Breakthrough Performance." *Google Research Blog*, April 6, 2022. Accessed December 20, 2024. https://research.google/blog/pathways-language-model-palm-scaling-to-540-billion-parameters-for-breakthrough-performance/.

[q] Artificial Analysis. Independent analysis of AI models and API providers AI. Accessed December 25, 2024. https://artificialanalysis.ai