Comparison of Major LLM Architectures (2017–2025)

Satyam Mittal LinkedIn

A concise, personal comparison of key LLM architectures developed over the past few years.
This document reflects my individual understanding and curiosity-driven research. This is by no means an exhaustive list, and many other excellent models exist in the field.

🎯 2017–2019

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
Transformer (2017) [1]	Encoder-Decoder Transformer	Multi-head self-attention (encoder & decoder) + cross-attention [1]	Fixed sinusoidal [1]	Post-layernorm [1]	ReLU [1]	~65M (base model) [1]	WMT14 translation corpora (e.g., 4.5M sentence pairs En→De) [1]	512 tokens [1]	Introduced self-attention to replace recurrent networks, enabling parallel sequence processing [1]	Supervised learning on translation tasks; residual connections, layer normalization, Adam optimizer [1]	Dramatically improved machine translation quality and speed; became foundational architecture for subsequent LLMs [1]
BERT (2018) [2]	Transformer Encoder (bidirectional)	Full bidirectional self-attention (MLM objective) [2]	Learned absolute [2]	Post-layernorm [2]	GELU [2]	110M (Base), 340M (Large) [2]	BooksCorpus + English Wikipedia (3.3B words total) [2]	512 tokens [2]	Masked Language Modeling and Next Sentence Prediction for deep bidirectional context understanding [2]	Unsupervised pre-training on large text corpus, then task-specific fine-tuning (transfer learning) [2]	Set new state-of-the-art on many NLP tasks (GLUE, QA) via contextualized embeddings and fine-tuning [2]
GPT (2018) [3]	Transformer Decoder (unidirectional)	Auto-regressive masked self-attention (causal LM) [3]	Learned absolute [3]	(Layernorm in transformer blocks) [3]	GELU [3]	117M [3]	BookCorpus (700M words of novels) [3]	512 tokens [3]	First to use generative pre-training for language understanding tasks, demonstrating transfer learning from unsupervised LM [3]	Unsupervised language model pre-training on unlabeled text, followed by supervised fine-tuning on each task [3]	Outperformed task-specific architectures on 9 of 12 NLP tasks via pre-trained knowledge, showing power of generative pre-training [3]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
GPT-2 (2019) [4]	Transformer Decoder (deep, uni-directional)	Masked multi-head self-attention (auto-regressive) [4]	Learned absolute [4]	(Layernorm in each layer) [4]	GELU [4]	1.5 billion [4]	WebText (8M web pages from Reddit links, ~40 GB) [4]	1024 tokens [4]	Demonstrated that much larger unsupervised language models can generate coherent long-form text [4]	Generative pre-training on vast internet text; no fine-tuning, evaluated zero-shot on tasks [4]	Achieved notable zero-shot performance on diverse tasks (QA, translation, summarization), indicating emergent multitask learning abilities [4]
XLNet (2019) [5]	Transformer-XL Decoder (autoregressive)	Permutation-based full self-attention (two-stream) [5]	Segment-aware relative positional encoding [5]	Post-layernorm [5]	GELU [5]	340M (Large) [5]	Diverse large text corpora (Google Books, Wikipedia, Giga5, ClueWeb, Common Crawl) [5]	512 tokens [5]	Generalized autoregressive pre-training that leverages all context positions (permuted order) instead of masking [5]	Memory-augmented Transformer (recurrence from Transformer-XL) with two-stream attention; trained with permutation language modeling objective [5]	Outperformed BERT on NLP benchmarks (e.g., GLUE) by capturing bidirectional context without an explicit mask, improving downstream task performance [5]

🎯 2020–2022

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
GPT-3 (2020) [6]	Transformer Decoder (very deep)	Masked multi-head self-attention (auto-regressive) [6]	Learned absolute (2048 tokens) [6]	Pre-layernorm [6]	GELU [6]	175 billion [6]	~300B tokens from Common Crawl, WebText2, Books, Wikipedia [6]	2048 tokens [6]	Massive scale showed emergent few-shot learning — model can perform tasks from prompts without fine-tuning [6]	Trained on extremely large corpus with mixed precision and model-parallelism across GPUs; no task-specific fine-tuning required for evaluation [6]	Achieved state-of-the-art in few-shot and zero-shot settings on many NLP tasks; demonstrated the benefits of scale for versatility [6]
T5 (2020) [7]	Transformer Encoder–Decoder	Full self-attention (enc & dec) + cross-attention [7]	Relative positional embeddings [7]	Pre-layernorm [7]	ReLU (with variants explored) [7]	11 billion (largest) [7]	C4 (Colossal Cleaned Common Crawl, ~750 GB text) [7]	512 tokens [7]	Unified “text-to-text” framework – model treats every NLP task (translation, QA, summarization, etc.) as text generation [7]	Unsupervised pre-training on C4 corpus with a denoising objective; followed by task-specific fine-tuning in a text-to-text format [7]	Achieved state-of-the-art on numerous benchmarks with one model applicable to all tasks; open-sourced in various sizes for flexible fine-tuning [7]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
Switch Transformer (2021) [8]	Transformer Decoder (Mixture-of-Experts)	Sparse MoE multi-head attention (experts in FFN layers) [8]	Learned absolute [8]	Pre-layernorm [8]	SwiGLU [8]	1.6 trillion (with 64 experts, ~26B active per token) [8]	C4 corpus (same as T5) [8]	2048 tokens [8]	Introduced conditional computation: uses routing to activate one expert feed-forward network per token, enabling extreme scale with efficient compute [8]	MoE training with load-balancing loss to ensure experts are utilized; scaled on TPU pods (Pathways) to reach trillion+ parameters [8]	Matched dense model quality with much lower computational cost; set new scale records (trillion+ parameters) while maintaining strong zero-shot and one-shot performance [8][9]
GLaM (2022) [9]	Transformer Decoder (Mixture-of-Experts)	Sparse mixture-of-experts (two experts per token) [9]	Learned absolute [9]	Pre-layernorm [9]	GELU [9]	1.2 trillion (64 experts, 2 active per token) [9]	Massive multilingual web corpus (filtered web pages, dialogues, code) [9]	2048 tokens [9]	Scaled MoE further with a balanced gating approach (each token routed to 2 experts) for efficiency – 7× parameter count of GPT-3 with 1/3 the energy cost [9]	Pre-trained with sparsely activated experts to reduce FLOPs; required specialized initialization and auxiliary losses for expert balance [9]	Outperformed GPT-3 in zero-/one-shot tasks while using significantly less inference compute per token; demonstrated efficient super-scaling of model capacity [9]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
Gopher (2021) [10]	Transformer Decoder (dense)	Multi-head self-attention (auto-regressive LM) [10]	Learned absolute [10]	Pre-layernorm [10]	GELU [10]	280 billion [10]	MassiveText dataset (multi-domain text: web, books, news, code) [10]	2048 tokens [10]	Systematic study of scaling up to 280B parameters with extensive evaluation on 152 tasks; highlighted strengths (knowledge recall) and weaknesses (logic, math) at scale [10]	Trained on TPU v3 Pod with mixed precision; used distributed training and periodic evaluation to analyze performance trends across model sizes [10]	Showed that increasing model size yields broad knowledge gains but plateaus on certain reasoning tasks, informing later research on data vs. model size trade-offs [10]
Chinchilla (2022) [11]	Transformer Decoder (dense)	Multi-head self-attention (auto-regressive LM) [11]	Learned absolute [11]	Pre-layernorm [11]	GELU [11]	70 billion [11]	1.4 trillion tokens of text (MassiveText, 4× Gopher's data) [11]	2048 tokens [11]	Established the compute-optimal model paradigm: a smaller model trained on more data can outperform a larger model trained on less data [11]	Used the same compute budget as Gopher but with 4× training tokens and a 4× smaller model, following new scaling law predictions [11]	Outperformed the 280B Gopher on many benchmarks despite far fewer parameters, demonstrating the importance of adequately scaling data quantity for a given model size [11]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
LaMDA (2022) [12]	Transformer Decoder (dialogue-optimized)	Multi-head self-attention (conversation LM) [12]	Learned absolute [12]	Pre-layernorm [12]	Swish (SiLU) [12]	137 billion [12]	1.56T words of public dialog data + web text (pre-training) [12]	2048 tokens [12]	Specialized for open-ended dialogue, with fine-tuning to improve safety and factual grounding in responses [12]	Pre-trained on dialog-heavy corpus, then fine-tuned with human-annotated data for safety; allowed to consult external tools/APIs during generation (to ground facts) [12]	Produced more engaging, contextually relevant, and safer conversational responses, marking a step toward AI that can hold human-like dialogue [12]
PaLM (2022) [13]	Transformer Decoder (dense)	Multi-head self-attention (auto-regressive LM) [13]	Rotary positional embedding [13]	Pre-layernorm [13]	SwiGLU [13]	540 billion [13]	780B tokens (multilingual web, books, GitHub code, conversations) [13]	2048 tokens [13]	Achieved breakthrough few-shot performance, exceeding human average on BIG-bench, and enabled strong multi-step reasoning and code generation [13]	Trained on Pathways system across TPU v4 Pods, leveraging mixed parallelism; incorporated multitask fine-tuning (FLAN) after pre-training for broad capabilities [13]	Set new state-of-the-art on many NLP benchmarks; demonstrated emergent abilities at scale (complex reasoning, coding, multilingual understanding) [13]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data	Context Length	Innovations	Training Strategies	Capabilities
InstructGPT / ChatGPT (2022) [14]	Transformer Decoder (GPT-3.5 series)	Masked multi-head self-attention (with instruction tuning) [14]	Learned absolute [6]	Pre-layernorm [6]	GELU [6]	175B (base model) [6]	GPT-3's pre-training data + human-generated dialogues and feedback data [14]	2048–4096 tokens [14]	Aligned language model with user intentions using Reinforcement Learning from Human Feedback (RLHF), greatly improving helpfulness and safety [14]	Supervised fine-tuning on demonstration data, then RLHF: model outputs rated by humans to train a reward model, and policy optimized via PPO [14]	Delivered far more user-friendly responses than raw GPT-3; reduced harmful outputs and followed instructions better, leading to ChatGPT's widespread adoption [14]

🎯 2023–2025

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data / Domain	Context Length	Innovations	Training Strategies	Capabilities
GPT-4 (2023) [15]	Transformer (dense, multimodal)	Multi-head self-attention (text & vision inputs) [15]	Enhanced positional encoding (8k–32k context) [15]	(Details not public) [15]	(Details not public) [15]	**Not disclosed** (estimated ≈1.8T, MoE architecture) [16]	Web text (pre-training); fine-tuned with code and imagery (multimodal) [15]	8,192 tokens (32,768 in extended version) [15]	Demonstrated powerful few-shot and reasoning abilities, with added vision input capability (accepts images as part of prompt) [15]	Post-trained with human feedback and model self-evaluation for alignment (Reinforcement Learning with human & AI feedback) [15]	Achieved top-level performance on a wide range of tasks (coding, math, vision-language understanding) and exams; significantly more reliable and creative than earlier models [15]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data / Domain	Context Length	Innovations	Training Strategies	Capabilities
LLaMA (2023) [17]	Transformer Decoder (open-source)	Multi-head self-attention (auto-regressive) [17]	Rotary positional embeddings (RoPE) [17]	RMSNorm (pre-normalization) [17]	SwiGLU [17]	7B–65B (65B largest) [17]	1.0T tokens of publicly available text (Common Crawl, Wikipedia, GitHub, etc.) [17]	2048 tokens [17]	Open-sourced high-performance foundation model, achieved GPT-3-level performance with 10× fewer parameters by efficient training and architecture tweaks [17]	Trained on curated large-scale dataset with extensive data cleaning and deduplication; utilized novel training efficiencies (such as mixed precision) [17]	Enabled broad research and downstream customization (e.g., fine-tuned chat models) due to open access; foundation for many derivative models (Alpaca, etc.), democratizing LLM research [17]
PaLM 2 (2023) [18]	Transformer Decoder (dense)	Multi-head self-attention (enhanced) [18]	ALiBi positional bias (longer context) [18]	Pre-layernorm [18]	GELU [18]	340B (reportedly, “Ultra” model) [18] *	Improved dataset spanning multiple languages, code, and math reasoning data [18]	4096 tokens [18]	More compute-efficient than PaLM with improved multilingual and reasoning skills; strong coding ability and domain expertise via focused training data [18]	Trained with an updated mixture of objectives (e.g., supervised learning on reasoning and coding tasks in addition to LM); leveraged prior PaLM insights with reduced parameter count [18]	Achieved superior performance across many benchmarks including logic and translation tasks; formed the backbone of Google’s Bard and enterprise models with faster inference [18]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data / Domain	Context Length	Innovations	Training Strategies	Capabilities
Claude (2023) [19]	Transformer Decoder (aligned AI)	Multi-head self-attention (with long-context support) [20]	Learned absolute (expanded context window) [20]	Pre-layernorm [19]	GELU [19]	52B (Claude 1) to 100B+ (Claude 2) [19] *	Conversational and knowledge domains (fine-tuned from a GPT-3.5-like base) [19]	100,000 tokens (Claude 2, extended context version) [20]	Pioneered “Constitutional AI” to align model behavior via AI feedback rather than only human feedback, yielding a safer yet minimally supervised assistant [19]	Initially fine-tuned with human feedback similar to InstructGPT, then optimized via a set of written principles (a “constitution”) that the AI uses to self-refine its answers [19]	Exhibits high-quality, less toxic dialogue and can handle extremely long documents in a single prompt (100k tokens), enabling analysis of lengthy texts; one of the first serious competitors to OpenAI’s models [20]
Gemini (2023) [21]	Multimodal Transformer (text, code, vision, audio)	Multi-modal self-attention integrating different data types [21]	Learned positional + modality-specific encodings [21]	Pre-layernorm [21]	SwiGLU [21]	>1T parameters [21]	Multimodal and multilingual dataset (web text, images, code, audio, video) [21]	128k tokens [22] *	Natively multimodal from the ground up—trained on text and other modalities together, enabling fluid combination of modalities and advanced reasoning abilities [21]	Pre-trained jointly on diverse modalities then fine-tuned with targeted multimodal datasets; incorporates tool use (e.g. search, APIs) and code execution during fine-tuning for “agentic” behavior [21]	Achieved state-of-the-art on vision-language and multimodal benchmarks; capable of complex reasoning and planning across text, images, and more, representing Google DeepMind’s answer to GPT-4 [21]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data / Domain	Context Length	Innovations	Training Strategies	Capabilities
Mistral (2023) [25]	Transformer Decoder (dense, efficient)	Sliding-window attention + grouped-query attention (GQA) [25]	Rotary positional embeddings (RoPE) [25]	RMSNorm [25]	SwiGLU [25]	7B [25]	Public domain English text, code, and reasoning tasks (OpenWeb, StackExchange, etc.) [25]	8,192 tokens [25]	Optimized for efficiency and performance; improved context window and inference cost compared to LLaMA 2 [25]	Trained on curated high-quality data with attention optimization and efficient token handling [25]	Open-weight model with strong results on open benchmarks and better inference efficiency on edge devices [25]
Qwen-1.5 (2024) [26]	Transformer Decoder (dense)	Multi-head self-attention + GQA [26]	RoPE + extended context window (up to 128K) [26]	RMSNorm [26]	SwiGLU [26]	0.5B – 72B [26]	Multilingual + code-heavy datasets, instruction tuning [26]	8k to 128k tokens [26]	Introduced a wide range of open models from lightweight to ultra-scale sizes with strong performance and multilingual support [26]	Instruction tuning, data deduplication, and large-context training pipelines for global use [26]	Versatile, competitive models across open benchmarks in both English and Chinese; notable open-source support via HuggingFace [26]
LLaMA 3 (2024) [27]	Transformer Decoder (dense, efficient)	Multi-head self-attention + GQA [27]	RoPE with longer context support (128k planned) [27]	RMSNorm [27]	SwiGLU [27]	8B – 65B (current), 400B+ planned [27]	Cleaned Common Crawl, Github, multilingual text, academic papers [27]	8k–128k tokens [27]	Meta's next-gen open models with improved alignment, multilingual performance, and code/data reasoning [27]	Fine-tuned on curated datasets with alignment objectives (chat, code, math), trained on Meta's Research SuperCluster [27]	Intended as GPT-4-class public alternative, with strong few-shot, multilingual, and tool-use capabilities [27]

Model (Year)	Architecture Type	Attention Type	Positional Encoding	Normalization	Activation	Parameters	Training Data / Domain	Context Length	Innovations	Training Strategies	Capabilities
DeepSeek-R1 (2025) [28] [29]	Transformer Decoder (Mixture-of-Experts)	Multi-head self-attention + MoE feed-forward (32 experts per layer) [28]	ALiBi positional bias (extremely long context) [28] *	Pre-layernorm [28]	GELU [28]	671 billion (MoE; ~37B parameters active per token) [28]	Broad web and knowledge corpora; specialized logical reasoning datasets [28]	128,000 tokens [28]	“Reasoning-centric” LLM optimized via large-scale reinforcement learning to excel at step-by-step problem solving and logic tasks, with unprecedented context length [28]	Multi-stage training: pretrained on diverse text, then purely reinforcement learning on reasoning tasks (no supervised fine-tune), plus reward-model guiding and distillation into smaller models [28]	Matches or surpasses similar-sized dense models on math, coding, and logic benchmarks at a fraction of training cost; open-sourced by a Chinese startup, sparking global competitive pressure in advanced AI capabilities [28]

🎯 References:

1. *Attention Is All You Need*, Ashish Vaswani et al., 2017 – NeurIPS. [Paper]
2. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, Jacob Devlin, Ming-Wei Chang, Kenton Lee, & Kristina Toutanova, 2018 – NAACL. [Paper]
3. *Improving Language Understanding by Generative Pre-Training*, Alec Radford, Karthik Narasimhan, Tim Salimans, & Ilya Sutskever, 2018 – OpenAI (Technical Report). [PDF]
4. *Language Models are Unsupervised Multitask Learners*, Alec Radford et al., 2019 – OpenAI (Technical Report). [PDF]
5. *XLNet: Generalized Autoregressive Pretraining for Language Understanding*, Zhilin Yang et al., 2019 – NeurIPS. [Paper]
6. *Language Models are Few-Shot Learners*, Tom B. Brown et al., 2020 – NeurIPS. [Paper]
7. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*, Colin Raffel et al., 2020 – J. Machine Learning Research. [Paper]
8. *Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity*, William Fedus, Barret Zoph, & Noam Shazeer, 2021 – JMLR. [Paper]
9. *GLaM: Efficient Scaling of Language Models with Mixture-of-Experts*, Nan Du et al., 2022 – ICML. [Paper]
10. *Scaling Language Models: Methods, Analysis & Insights from Training Gopher*, Jack W. Rae et al., 2021 – DeepMind (Technical Report). [Paper]
11. *Training Compute-Optimal Large Language Models*, Jordan Hoffmann et al., 2022 – DeepMind (NeurIPS). [Paper]
12. *LaMDA: Language Models for Dialog Applications*, Romal Thoppilan et al., 2022 – arXiv Preprint. [Paper]
13. *PaLM: Scaling Language Modeling with Pathways*, Aakanksha Chowdhery et al., 2022 – arXiv Preprint. [Paper]

14. *Training language models to follow instructions with human feedback*, Long Ouyang et al., 2022 – OpenAI (NeurIPS). [Paper]
15. *GPT-4 Technical Report*, OpenAI, 2023. [Paper]
16. *GPT-4 has more than a trillion parameters – Report*, Matthias Bastian, 2023 – The Decoder. [Article]
17. *LLaMA: Open and Efficient Foundation Language Models*, Hugo Touvron et al., 2023 – Meta AI. [Paper]
18. *PaLM 2 Technical Report*, Rohan Anil et al., 2023 – Google. [Paper]
19. *Constitutional AI: Harmlessness from AI Feedback*, Yuntao Bai et al., 2022 – Anthropic. [Paper]
20. *Introducing 100K Context Windows*, Anthropic, 2023. [Blog]
21. *Gemini: A Family of Highly Capable Multimodal Models* (Technical Report), Google DeepMind, 2023. [PDF]
22. **Discover AI** - Youtube channel
23. Mistral 7B Technical Report, Mistral AI, 2023. [Blog]
24. Qwen: The Qwen-1.5 Series, Alibaba DAMO Academy, 2024. [HuggingFace]
25. LLaMA 3 Preview, Meta AI, 2024. [Blog]
26. *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*, DeepSeek-AI, 2025. [Paper]
27. *DeepSeek-R1 model now available in Amazon Bedrock Marketplace...*, Vivek Gangasani, Banu Nagasundaram, Jonathan Evans, & Niithiyn Vijeaswaran, 2025 – AWS Blog. [Article]

Note:

- The LLMs listed here represent a personal selection from the many published models in recent years, based on what I've studied or followed. This is not an exhaustive list, and many excellent models may not be included here.
- This document is designed to keep me updated with growing LLM architectures. It is based on my personal understanding and synthesis of various research papers.
- If you spot any errors or have suggestions for improvement, please feel free to reach out.
- The references mentioned above have been explored over the past few months. There may be more detailed explanations available, and I'd be happy to dig deeper into those if needed.
- Special thanks to ChatGPT for helping with formatting and all the nitty-gritty details etc. 😃