Nimse Vidya Kisan, 2026, 14:5
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Large Language Models: The Core of Generative AI — A Review Of Architecture, Training, Capabilities and Limitations

Nimse Vidya Kisan[1] Prof. J. M. Patade[2] Prof. M. C. Kolse[3] Prof. S. K. Said[4]
Department of Artificial Intelligence and Data Science
Jaihind College of Engineering
Pune, Maharashtra, India

Abstract—Large Language Models (LLMs) have become the core technology of Generative Artificial Intelligence, powering chat assistants, writing and coding tools, search and many other applications. This paper reviews how LLMs work and why they matter. It explains the transformer architecture with self-attention, tokenization into subword units and the decoder-only design used by generative models, then traces the development from generative pre-training through models that perform tasks from a few examples. Scaling laws and compute-optimal training explain why larger models trained on more data perform better, and emergent abilities and chain-of-thought prompting show how capabilities appear with scale. Alignment through instruction tuning and reinforcement learning from human feedback, and grounding through retrieval, are reviewed. Open model families are discussed. A five-stage LLM lifecycle architecture, comprising data, pre-training, alignment, augmentation and application, is proposed. Research gaps, objectives, a comparison of model families and limitations including hallucination, bias, cost, environmental impact, privacy and misuse are analysed. The review concludes that LLMs are a foundational general-purpose technology whose benefits depend on responsible development, grounding and human oversight.

Index Terms—Large language models, generative AI, transformer, self-attention, pre-training, scaling laws, RLHF, prompt engineering

I. INTRODUCTION

Within a few years, generative AI moved from research labs to everyday tools used by students, programmers, writers and businesses. At the centre of this change are large language models: neural networks with billions of parameters trained to predict the next word on vast collections of text. Surprisingly, this simple objective produces models that can answer questions, summarize, translate, write code and reason through problems.

Zhao *et al.* [1] surveyed the development of LLMs, and Bommasani *et al.* [2] described them as foundation models: general models adapted to many downstream tasks. Understanding how they are built, what they can do and where they fail is now part of basic AI literacy.

This paper reviews the architecture, training, capabilities and limitations of LLMs and proposes a lifecycle architecture.

II. BACKGROUND AND RELATED WORK

1. The Transformer

Vaswani *et al.* [3] introduced the transformer, which replaces recurrence with self-attention so that every token can attend to every other token in parallel. Text is split into subword tokens using methods such as byte-pair encoding [4]. Fig. 1 summarizes LLM components and the stages of their lifecycle.

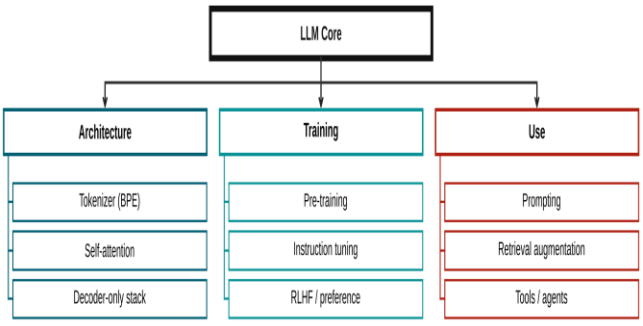


Fig. 1. Components and lifecycle stages of large language models.

© 2026 Nimse Vidya Kisan,, This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

2. Model Types

Encoder models such as BERT [5] are suited to understanding tasks, while decoder-only models generate text autoregressively and form the basis of generative assistants.

3. Review Of Related Studies

Radford *et al.* [6] showed that generative pre-training of a transformer followed by fine-tuning improves many language tasks, and GPT-2 [7] demonstrated that larger models perform tasks without task-specific training. Brown *et al.* [8] introduced GPT-3 and showed few-shot learning from examples in the prompt.

Kaplan *et al.* [9] found that loss falls predictably as a power law with model size, data and compute, and Hoffmann *et al.* [10] refined this into compute-optimal scaling, showing that many models were under-trained. Wei *et al.* [11] documented emergent abilities that appear only above certain scales, and chain-of-thought prompting [12] elicited step-by-step reasoning.

For alignment, Christiano *et al.* [13] learned from human preferences, and Ouyang *et al.* [14] used reinforcement learning from human feedback to make models follow instructions helpfully. Touvron *et al.* [15] released openly available LLaMA models. Lewis *et al.* [16] grounded generation in retrieved documents. Bender *et al.* [17] warned about the risks of ever-larger models, including bias, environmental cost and the illusion of understanding. Table I summarizes these works.

Sr. No.	Ref.	Year	Methodology	Findings	Research Gap
1	[3]	2017	Transformer self-attention	Parallel sequence modelling	Quadratic attention cost
2	[6]	2018	Generative pre-training	Pre-train + fine-tune gains	Small by today's standards
3	[8]	2020	GPT-3 few-shot learning	In-context learning	Hallucination and bias
4	[9]	2020	Scaling laws	Predictable improvement with scale	Compute allocation refined later
5	[17]	2021	Risks of large models	Bias, cost, misunderstanding	Position paper
6	[10]	2022	Compute-optimal training	More data per parameter	Large training cost
7	[11]	2022	Emergent abilities	Capabilities appear with scale	Measurement debated
8	[14]	2022	RLHF instruction following	Helpful, aligned behaviour	Labeller cost

TABLE I Summary of Literature Survey

III. RESEARCH GAP AND OBJECTIVES

1. Identified Research Gaps

1) LLMs hallucinate and can state false information confidently [18].

1. Training data contain social biases that models may reproduce.

2. Training and running large models consume substantial energy and money.

3. Support for low-resource and Indian languages is uneven.

4. Evaluation of real-world reliability and safety remains immature.

2. Objectives Of The Review

- To explain the transformer architecture and tokenization underlying LLMs.
- To review pre-training, scaling and emergent capabilities.
- To examine alignment and retrieval augmentation.
- To propose an LLM lifecycle architecture.

- To analyse limitations and responsible-use considerations.

IV. DESIGN OF THE PROPOSED SYSTEM

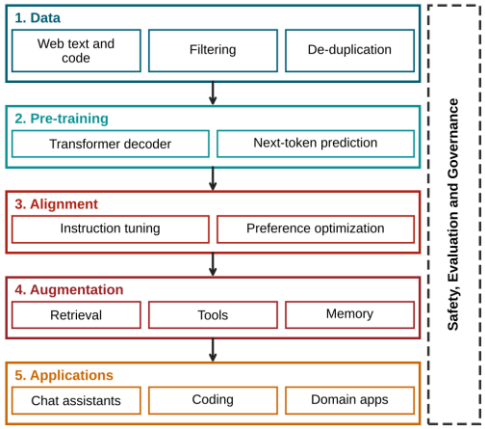


Fig. 2 shows the lifecycle architecture of an LLM-based generative AI system.

1. Data Layer

Large, diverse text and code corpora are collected, filtered for quality, de-duplicated and cleaned of personal information.

2. Pre-Training Layer

A decoder-only transformer is trained to predict the next token over trillions of tokens on large accelerator clusters.

3. Alignment Layer

Instruction tuning and preference optimization teach the model to follow instructions, be helpful and avoid harmful outputs.

4. Augmentation Layer

Retrieval, tools and memory connect the model to current information and external actions.

5. Application Layer

Chat assistants, coding tools, summarizers and domain applications use the model through APIs, with monitoring and feedback.

6. Methodology

Fig. 3 outlines how an LLM produces a response. Self-attention computes

$$Attention(Q,K,V) = softmax(QK^T / \sqrt{d_k})\ V \quad (1)$$

and pre-training minimizes the negative log-likelihood of each next token,

$$L = -\Sigma_t \log P(x_t \mid x_1 \ldots x_{t-1}) \quad (2)$$

At inference, tokens are generated one at a time by sampling from the predicted distribution, controlled by temperature and other decoding settings.

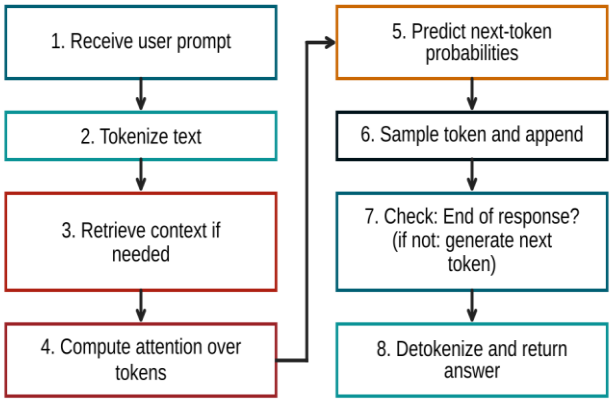


Fig. 3. How an LLM generates a response.

V. DISCUSSION AND COMPARISON

Table II compares model families. Encoder models excel at understanding, decoder-only models at generation, closed frontier models offer the highest capability, and open models allow local deployment and customization.

Family	Strength	Typical Use	Deployment
Encoder (BERT)	Understanding	Classification, search	Easy, small
Encoder–decoder (T5)	Text-to-text	Translation, summarization	Moderate
Decoder-only (GPT-style)	Generation	Chat, writing, coding	Large compute
Open models (LLaMA)	Customizable	Local / domain apps	Self-hosted

TABLE II Comparison of Language Model Families

VI. CHALLENGES AND FUTURE DIRECTIONS

Table III summarizes challenges. Future directions include more efficient architectures, multimodal models, better reasoning and factuality, Indian language models and governance frameworks for safe deployment.

Challenge	Mitigation
Hallucination	Retrieval grounding; verification
Bias	Data curation; evaluation; alignment
Cost and energy	Efficient models; quantization
Privacy	Data filtering; on-premise deployment
Misuse	Usage policies; safety filters

TABLE III Challenges and Mitigation

VII. CONCLUSIONS

Large language models are the engine of generative AI. The reviewed literature shows how the transformer, scale and alignment combine to produce general-purpose systems capable of a wide range of language tasks. Their limitations, including hallucination, bias, cost and misuse, mean that LLMs should be grounded, evaluated and used with human oversight to deliver their benefits responsibly.

1. Acknowledgment

The authors express their sincere gratitude to the Department of Artificial Intelligence and Data Science, Jaihind College of Engineering, Pune, for providing the facilities, guidance and encouragement needed to carry out this work. The authors also thank the faculty members and colleagues whose valuable suggestions helped improve the quality of this review.

REFERENCES

1. W. X. Zhao et al., "A survey of large language models," arXiv:2303.18223, 2023.
2. R. Bommasani et al., "On the opportunities and risks of foundation models," arXiv:2108.07258, 2021.
3. A. Vaswani et al., "Attention is all you need," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, 2017.
4. R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in Proc. 54th Annu. Meeting ACL, 2016, pp. 1715–1725.
5. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
6. A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," OpenAI, Tech. Rep., 2018.
7. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," OpenAI, Tech. Rep., 2019.
8. T. Brown et al., "Language models are few-shot learners," in Proc. NeurIPS, vol. 33, 2020, pp. 1877–1901.
9. J. Kaplan et al., "Scaling laws for neural language models," arXiv:2001.08361, 2020.
10. J. Hoffmann et al., "Training compute-optimal large language models," in Proc. NeurIPS, vol. 35, 2022, pp. 30016–30030.
11. J. Wei et al., "Emergent abilities of large language models," Trans. Mach. Learn. Res., 2022.
12. J. Wei et al., "Chain-of-thought prompting elicits reasoning in large language models," in Proc. NeurIPS, vol. 35, 2022, pp. 24824–24837.
13. P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in Proc. NeurIPS, vol. 30, 2017.
14. L. Ouyang et al., "Training language models to follow instructions with human feedback," in Proc. NeurIPS, vol. 35, 2022, pp. 27730–27744.
15. H. Touvron et al., "LLaMA: Open and efficient foundation language models," arXiv:2302.13971, 2023.
16. P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Proc. NeurIPS, vol. 33, 2020, pp. 9459–9474.
17. E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?" in Proc. ACM Conf. Fairness, Accountability, Transparency (FAccT), 2021, pp. 610–623.

18. Z. Ji et al., "Survey of hallucination in natural language generation," ACM Comput. Surv., vol. 55, no. 12, Art. no. 248, 2023.