The Technological Evolution of Large Language Models: From Attention Mechanisms to Llama 4

Michael Gathara, ChatGPT Deep Research & Gemini Deep Research

Introduction

Large Language Models (LLMs) have emerged as a pivotal technology in the field of artificial intelligence, demonstrating remarkable capabilities in understanding and generating human language. Their development represents a significant leap forward in natural language processing, enabling applications ranging from sophisticated chatbots to advanced text summarization and code generation. A foundational moment in this evolution was the introduction of the Transformer architecture in the 2017 research paper "Attention is All You Need".[1] This paper proposed a novel approach to sequence transduction that has since become the cornerstone of most modern LLMs.[1] By dispensing with the complex recurrent and convolutional neural networks that were prevalent at the time [3], the Transformer architecture paved the way for unprecedented advancements in the field. This report aims to chronologically trace the technological journey of LLMs, starting from the introduction of the Transformer architecture and culminating in the architecture of the recently unveiled Llama 4 model, highlighting the key innovations and their impact at each stage.

The Foundation: "Attention is All You Need" (2017)

Key Innovations: The Transformer Architecture

The "Attention is All You Need" paper introduced the Transformer, a groundbreaking neural network architecture that shifted away from reliance on recurrent and convolutional layers, instead leveraging attention mechanisms as its core building blocks.[2] The Transformer model is fundamentally comprised of an encoder and a decoder, both constructed from stacks of self-attention layers and point-wise, fully connected feed-forward networks.[3] A primary reason for the widespread adoption of this architecture in modern LLMs is its inherent parallelizability.[1] Unlike its sequential processing predecessors such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, the Transformer can process all parts of the input sequence simultaneously, leading to significantly faster training times.[1] The architecture also incorporates input embeddings to capture the semantic meaning of individual tokens and positional encoding to provide the model with information about the order of these tokens within the sequence.[2] To facilitate the training of deeper and more complex networks, residual connections and layer normalization are employed around each sub-layer within the encoder and decoder.[2] In the decoder component,

masked self-attention is utilized to ensure that during the generation process, the model only attends to previously generated tokens, thus maintaining the crucial autoregressive property required for language generation.[3] Furthermore, the decoder incorporates an encoder-decoder attention mechanism that allows it to focus on the most relevant parts of the input sequence as it generates the output.[3] This design allows the Transformer to effectively model dependencies between words in a sequence, regardless of their distance, a significant improvement over the challenges faced by recurrent models in capturing long-range dependencies.[3]

The Self-Attention Mechanism: Scaled Dot-Product and Multi-Head Attention

At the heart of the Transformer's capabilities lies the self-attention mechanism, which enables the model to weigh the importance of different tokens within the input sequence, thereby capturing intricate relationships and dependencies between them.[3] The specific form of self-attention employed is the scaled dot-product attention.[1] This mechanism calculates attention scores by taking the dot product of three learned representations of the input sequence: the query, key, and value vectors. These dot products, which represent the similarity between each query and key, are then scaled by the square root of the dimension of the key vectors to prevent them from becoming too large, which could destabilize the training process.[2] A softmax function is subsequently applied to these scaled scores, normalizing them into weights that indicate the amount of attention each query should pay to the corresponding key.[2] Finally, these attention weights are used to compute a weighted sum of the value vectors, producing the output of the attention mechanism for each position in the input sequence.[2] To further enhance the model's ability to capture diverse relationships, the paper introduced multi-head attention.[1] Instead of using a single attention function, multi-head attention employs multiple parallel attention heads. Each head learns its own independent linear projections of the query, key, and value matrices, allowing the model to simultaneously focus on different aspects of the relationships between words in the sequence.[1] The outputs from all these parallel attention heads are then concatenated and passed through a final linear transformation to produce the ultimate output of the multi-head attention layer.[1] This mechanism not only improves the model's accuracy but also contributes to the parallelizability of the architecture.[18]

Dispensing with Recurrence and Convolutions: Advantages and Implications

The Transformer architecture distinguished itself as the first sequence transduction model to rely solely on self-attention, completely eliminating the need for recurrence (as in RNNs) and convolutions.[3] This departure from previous approaches brought several key advantages. Unlike recurrent layers, which process sequences sequentially

and thus have a computational complexity that scales linearly with the sequence length, a self-attention layer can connect all positions within the sequence with a constant number of sequentially executed operations per layer.[3] Furthermore, the inherent sequential nature of recurrent models limits their ability to parallelize computations within training examples. In contrast, the Transformer architecture allows for significantly more parallelization, as the computation for each position in the sequence can be performed independently.[1] This parallel processing capability is particularly beneficial for leveraging the power of GPUs, leading to faster training and inference times.[6] Additionally, the Transformer's self-attention mechanism enables it to handle longer sequences more effectively than RNNs.[3] In RNNs, information from earlier parts of the sequence can be diluted or forgotten as the sequence length increases, a phenomenon known as the vanishing gradient problem.[11] The Transformer, however, with its direct connections between all tokens facilitated by self-attention, provides a more robust mechanism for capturing and retaining information across long sequences.[3] The success of the Transformer architecture in machine translation [3] demonstrated its potential as a versatile model for sequence transduction tasks, paving the way for its widespread adoption and adaptation in various other natural language processing applications [6] and even in domains beyond NLP.[6]

The First Generation of Transformer-Based LLMs: Laying the Groundwork

GPT-1 (2018): Generative Pre-training and the Decoder-Only Approach

Following the introduction of the Transformer architecture, OpenAI developed GPT-1 (Generative Pre-trained Transformer 1) in 2018, marking the first in their series of large language models.[20] GPT-1 introduced the concept of generative pre-training, a self-supervised learning technique specifically tailored for natural language processing tasks.[21] This approach involves training a model to predict the next word in a sequence, allowing it to learn the underlying structure and patterns of language from vast amounts of unlabeled text data.[21] The architecture of GPT-1 was based on a 12-layer decoder-only Transformer.[20] This decoder comprised twelve masked self-attention heads, with each head operating on 64-dimensional states, resulting in a total of 768 dimensions per token.[20] The model was pre-trained on a substantial corpus of text consisting of approximately 7,000 unpublished fiction books, known as BookCorpus.[20] This extensive pre-training phase enabled GPT-1 to acquire a deep understanding of language structure and context.[21] After pre-training, the model was fine-tuned on various downstream tasks using labeled data specific to those tasks.[20] Despite minimal changes to its underlying task-agnostic architecture during

fine-tuning, GPT-1 achieved significant improvements over previous state-of-the-art models on several diverse language processing tasks, including natural language inference, question answering, commonsense reasoning, semantic similarity assessment, and text classification.[20] Unlike the original Transformer, which had both an encoder and a decoder, GPT-1 utilized only the decoder component.[25] This decoder-only architecture, coupled with the masked self-attention mechanism, was specifically designed for autoregressive language modeling, where the model sequentially predicts the next word based on the preceding words.[22] Furthermore, GPT-1 employed learned position embeddings to inform the model about the position of each token in the sequence, rather than using the fixed sinusoidal positional encodings proposed in the original Transformer paper.[23]

BERT (2018): Bidirectional Encoders and Masked Language Modeling

Also in 2018, Google introduced BERT (Bidirectional Encoder Representations from Transformers), a model that revolutionized language understanding by considering the context of a word from both the words that precede it and the words that follow it.[7] This bidirectional approach contrasted with the unidirectional processing of models like GPT-1.[25] Architecturally, BERT employs a multi-layer bidirectional Transformer encoder.[26] This encoder allows each token in the input sequence to attend to all other tokens, enabling it to capture contextual information from both directions.[29] BERT was pre-trained simultaneously on two novel tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP).[26] The MLM task involves randomly masking a certain percentage of words in the input sentence and training the model to predict the original masked words based on the context provided by the unmasked words.[26] This objective forces the model to learn deep bidirectional representations of language.[26] The NSP task trains the model to predict whether one sentence logically follows another, which helps in understanding relationships between sentences.[26] For its input representation, BERT uses WordPiece embeddings, which break words into subword units, and has a vocabulary of 30,000 tokens.[26] Additionally, BERT incorporates token type embeddings to distinguish between different segments of input, position embeddings to understand the order of tokens, and segment type embeddings to differentiate between multiple sentences in a single input sequence.[26] The pre-trained BERT model is designed to be a general-purpose language representation model that can be fine-tuned for a wide array of downstream NLP tasks by adding just one additional output layer, without requiring significant modifications to the core architecture.[26] The original BERT paper reported results for two model sizes: BERTBASE, which has 12 layers and 110 million parameters, and BERTLARGE, which has 24 layers and 340 million parameters.[26]

Architectural Differences and Distinct Pre-training Objectives

The primary architectural distinction between GPT-1 and BERT lies in their utilization of the Transformer framework. GPT-1 employs a decoder-only stack with masked self-attention, making it inherently unidirectional and focused on generating text sequentially.[25] In contrast, BERT utilizes an encoder-only stack with bidirectional self-attention, enabling it to process the entire input sequence at once and understand the context of each word based on both its preceding and succeeding words.[25] This fundamental architectural difference is closely tied to their distinct pre-training objectives. GPT-1 is pre-trained using a language modeling objective, where the goal is to predict the next word in a given sequence.[22] This objective naturally aligns with its decoder-only architecture and its strength in generative tasks. BERT, on the other hand, is pre-trained with the Masked Language Model (MLM) and Next Sentence Prediction (NSP) objectives.[25] The MLM objective trains the model to understand the context of words within a sentence by predicting masked words, while the NSP objective helps it understand the relationship between pairs of sentences. These pre-training tasks, combined with its bidirectional encoder architecture, make BERT particularly well-suited for language understanding tasks such as text classification, named entity recognition, and question answering.[10]

Model	Architecture	Parameters	Key Pre-training Objective(s)
GPT-1 (2018)	Decoder-only	117 million	Next word prediction
BERT Base (2018)	Encoder-only	110 million	Masked Language Model, Next Sentence Prediction
BERT Large (2018)	Encoder-only	340 million	Masked Language Model, Next Sentence Prediction
GPT-2 (2019)	Decoder-only	1.5 billion	Next word prediction
GPT-3 (2020)	Decoder-only	175 billion	Next word prediction (with modifications for sparse attention)

Scaling and Specialization: The Rise of Larger Models

GPT-2 (2019): Scaling Parameters and Emergent Language Capabilities

In 2019, OpenAI introduced GPT-2, a model that demonstrated the power of scaling up the size of language models.[50] GPT-2 was essentially a larger version of GPT-1, featuring a ten-fold increase in both its parameter count, reaching 1.5 billion, and the size of its training dataset, which consisted of 8 million web pages curated from upvoted links on Reddit and was named WebText.[50] GPT-2 retained the decoder-only Transformer architecture of its predecessor but incorporated some modifications aimed at improving performance and stability, such as moving layer normalization to the input of each sub-block and adding an additional layer normalization after the final self-attention block.[50] A significant finding with GPT-2 was the emergence of impressive language capabilities, including the ability to generate coherent and contextually rich text across a wide range of topics.[50] Remarkably, it could perform tasks like translation, question answering, summarization, and even generate different styles of text output without being explicitly fine-tuned for those specific tasks, showcasing the power of zero-shot learning.[50] The model was released in various sizes, including small, medium, large, and extra-large versions [51], and featured an expanded vocabulary of 50,257 tokens and a larger context size of 1024 tokens compared to GPT-1.[50] Its ability to generate thematically appropriate text, even for surreal prompts, highlighted its flexibility and the potential of large-scale language models.[50]

GPT-3 (2020): Massive Scale and Few-Shot Learning

Building upon the success of GPT-2, OpenAI released GPT-3 in 2020, pushing the boundaries of language model scaling even further.[55] GPT-3 was an unprecedentedly large model with 175 billion parameters, trained on a massive and diverse dataset of 45 terabytes of text data. This dataset comprised a wide range of sources, including Common Crawl, WebText2, Books1, Books2, and Wikipedia.[55] The architecture of GPT-3 was similar to GPT-2, retaining the decoder-only Transformer structure but incorporating modifications to accommodate its much larger scale, such as the use of alternating dense and locally banded sparse attention patterns, similar to the Sparse Transformer.[55] A key characteristic of GPT-3 was its strong "zero-shot" and "few-shot" learning abilities.[56] It could often achieve performance levels competitive with or even surpassing fine-tuned state-of-the-art models on various NLP datasets without requiring any gradient updates or explicit fine-tuning. Instead, tasks and a few examples demonstrating the desired behavior were provided purely through text interaction with the model.[56] GPT-3 demonstrated its versatility by performing a wide

array of tasks, including translation, question answering, cloze tasks, and even tasks that demanded on-the-fly reasoning or adaptation to new domains, all through simple text-based prompts.[59] Like GPT-2, GPT-3 was also available in different sizes, each with varying parameter counts, such as ada, babbage, curie, and davinci, offering different levels of capability and computational cost.[56]

Other Notable Early Transformer Models and their Contributions

Beyond the GPT family, other significant Transformer-based models emerged during this early period, each contributing unique advancements to the field. ELMo (Embeddings from Language Models), introduced in 2018, was a bi-directional LSTM-based model that generated contextualized word embeddings.[11] While not purely Transformer-based, ELMo represented an important step towards capturing word meaning in context, which later became a key strength of Transformer models. T5 (Text-to-Text Transfer Transformer), developed in 2019, adopted a unique approach by framing all natural language processing tasks as a text-to-text problem.[28] This unified framework demonstrated the versatility of the Transformer architecture by showing that a single model could perform a wide range of tasks simply by changing the input prompt. Also in 2019, XLNet was introduced, aiming to combine the benefits of both BERT's bidirectional context understanding and GPT's autoregressive generation capabilities through a novel permutation-based pre-training approach.[28] RoBERTa (Robustly Optimized BERT approach), also from 2019, was an enhanced version of BERT that achieved improved performance on several NLP benchmarks by training on a larger dataset for a longer duration and with a modified pre-training procedure.[64] Finally, DistilBERT, released in 2019 by Hugging Face, was a distilled and more efficient version of BERT.[47] It demonstrated that smaller models, trained using techniques like knowledge distillation, could retain a significant portion of the performance of their larger counterparts while being faster and requiring less memory.

Alternative Architectures: Exploring Beyond the Transformer

Recurrent Neural Networks (RNNs) and their Historical Significance in Language Modeling

Before the advent of the Transformer architecture, Recurrent Neural Networks (RNNs), including their more sophisticated variants like Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs), were the dominant models for sequence modeling and generation tasks in natural language processing.[8] RNNs are designed to process sequential data by maintaining an internal state, often referred to as a hidden state, which acts as a memory of the sequence history up to the current point.[8] This

recurrent nature allows them to capture dependencies between elements in a sequence, making them suitable for tasks like language modeling where the prediction of the next word depends on the preceding words. LSTMs, introduced to address the limitations of traditional RNNs in learning long-range dependencies due to the vanishing gradient problem, incorporated memory cells and gating mechanisms to control the flow of information over extended sequences.[11] Bidirectional RNNs further enhanced the context understanding by processing the input sequence in both forward and backward directions, allowing the model to consider information from both the past and the future when making predictions.[11] Despite these advancements, RNNs faced significant challenges, particularly in terms of parallel processing. Their inherent sequential nature made it difficult to efficiently train them on parallel computing architectures like GPUs.[10] Furthermore, RNNs often struggled to effectively model very long sequences, as the information from earlier parts of the sequence could become diluted or lost over many time steps.[10] These limitations ultimately paved the way for the development and widespread adoption of the Transformer architecture.

Convolutional Neural Networks (CNNs) and their Applications in NLP

While primarily renowned for their success in computer vision tasks, Convolutional Neural Networks (CNNs) have also found valuable applications within the field of natural language processing.[6] In NLP, CNNs are particularly effective at tasks such as text classification, sentiment analysis, and topic categorization.[6] They achieve this by applying convolutional filters over word embeddings, allowing them to capture hierarchical patterns within the text data.[75] These filters can learn to detect local features, such as n-grams or specific word combinations, and then combine these features in subsequent layers to understand more complex semantic structures.[75] CNNs are generally more efficient for tasks where capturing local dependencies and extracting specific features are crucial.[6] Recognizing the complementary strengths of different architectures, researchers have also explored hybrid models that combine Transformers with CNNs, particularly for tasks like image captioning and visual question answering, where both understanding sequential data (text) and processing spatial data (images) are required.[99]

Emerging Alternative Architectures and their Potential (e.g., Mamba)

While the Transformer architecture has become dominant for many large language models, ongoing research continues to explore alternative architectures that might address some of its computational limitations, especially when dealing with extremely long sequences of data.[99] One such promising alternative is Mamba, a state-space model (SSM) that has demonstrated compelling results in language modeling tasks.[100]

Unlike Transformers, which have a quadratic computational complexity with respect to the sequence length due to the attention mechanism, Mamba offers the potential for linear scaling, leading to faster inference times and lower computational costs, particularly for very long contexts.[100] Mamba achieves this efficiency by using a selective state-space mechanism that allows the model to focus on relevant information while processing sequences.[100] Other emerging alternative architectures include Linear RNNs, which aim to improve the efficiency of recurrent models, and hybrid approaches that seek to combine the strengths of Transformers with other types of neural networks.[99] These alternative architectures represent an active area of research, with the potential to offer more efficient and scalable solutions for future generations of large language models.

The Llama Lineage: Building on Established Success

Llama 1 (2023): Architectural Details, Training Data, and Methodology

In 2023, Meta introduced Llama 1 (Large Language Model Meta AI), a family of open-source large language models designed to promote research and democratize access to this technology.[80] The initial release included models ranging in size from 7 billion to 65 billion parameters.[76] Llama 1 was built upon the foundation of the Transformer architecture but incorporated several key improvements.[76] These included using RMSNorm (Root Mean Square Layer Normalization) for pre-normalization, which is computationally more efficient and improves training stability.[76] The ReLU (Rectified Linear Unit) non-linearity was replaced with the SwiGLU activation function, which has been shown to improve performance.[76] Additionally, Llama 1 removed absolute positional embeddings and instead utilized Rotary Positional Embeddings (RoPE) at each layer of the network. RoPE introduces rotation operations into the positional encoding process, allowing the model to learn dynamic positional representations.[76] The Llama 1 models were trained on a massive dataset comprising 1.4 trillion tokens, sourced exclusively from publicly available data, including web scrapes from CommonCrawl, open-source code repositories from GitHub, Wikipedia in multiple languages, public domain books from Project Gutenberg, the Books3 dataset, scientific papers from ArXiv, and questions and answers from Stack Exchange websites.[81] The development team focused on scaling the model's performance primarily by increasing the volume of training data rather than solely the number of parameters, recognizing that inference cost is a significant factor for LLMs.[84]

Llama 2 (2023): Key Improvements, Grouped-Query Attention, and Open-Source Impact

In July 2023, Meta, in partnership with Microsoft, announced Llama 2, the next generation of their large language model series.[83] Llama 2 offered significant enhancements over its predecessor in terms of scale, efficiency, and overall performance.[83] The models were trained on a dataset containing 2 trillion tokens, representing a 40% increase in the amount of training data compared to Llama 1.[84] Furthermore, the context length, which determines the amount of information the model can process at once, was doubled to 4096 tokens.[84] A key addition to the Llama 2 family was the release of not only foundational pre-trained models but also instruction fine-tuned chat models specifically optimized for dialogue use cases.[83] Larger models within the Llama 2 series, specifically the 34 billion and 70 billion parameter versions, incorporated Grouped-Query Attention (GQA).[77] GQA is a variant of the multi-head attention mechanism designed to improve inference scalability by reducing computational redundancy. It achieves this by sharing the key and value projections across multiple query heads, leading to more efficient processing, especially for longer sequences.[88] In a significant departure from the more restrictive access to Llama 1, all Llama 2 models were released with their weights and could be used for a wide range of commercial applications, further amplifying their impact on the open-source AI landscape.[84] This open and commercially permissive release fostered a more collaborative environment for developers and researchers worldwide, accelerating innovation in the field.[83]

Llama 3 (2024): Further Scaling, Enhanced Tokenizer, and Training Innovations

In 2024, Meta unveiled Llama 3, the latest iteration in their series of large language models, featuring 8 billion and 70 billion parameter versions.[84] Llama 3 demonstrated further improvements in reasoning abilities and achieved state-of-the-art performance on a wide range of industry benchmarks.[84] The models were pre-trained on an even larger dataset of over 15 trillion tokens, representing a seven-fold increase compared to Llama 2, and included four times more code data in the training mix.[109] A significant architectural enhancement in Llama 3 was the adoption of a new tokenizer with a vocabulary size of 128,000 tokens.[109] This larger vocabulary allows for more efficient encoding of language, which contributed to the improved performance of the models.[109] Similar to Llama 2, both the 8 billion and 70 billion parameter models in the Llama 3 family utilized Grouped-Query Attention (GQA) to enhance inference efficiency and scalability.[108] To prepare for future multilingual applications, the pre-training dataset for Llama 3 included over 5% of high-quality non-English data, covering more than 30 languages.[109] The training process for Llama 3 employed a "4D parallelism" strategy, encompassing data, model, pipeline, and context parallelism, to efficiently distribute the computational workload across a large number of GPUs.[108] Furthermore, the post-training phase involved a combination of Supervised

Fine-Tuning (SFT), Rejection Sampling (RS), and Direct Preference Optimization (DPO) techniques to align the model's behavior with human preferences for helpfulness and safety.[110]

The Cutting Edge: Llama 4 (2025)

Reported Architecture and Technical Details

The latest advancement in Meta's large language model series is Llama 4, which was unveiled in 2025.[115] A significant architectural innovation in Llama 4 is the introduction of a "mixture of experts (MoE) architecture".[115] In MoE models, only a specific subset of the model's total parameters, referred to as "experts," are activated for any given input token. This approach allows for increased model capacity and improved efficiency during both training and inference.[115] The initial release of Llama 4 included two primary models: Llama 4 Scout, featuring 17 billion active parameters out of a total of 109 billion parameters and utilizing 16 experts, and Llama 4 Maverick, also with 17 billion active parameters but with a total of 400 billion parameters distributed across 128 experts.[115] A key feature of Llama 4 is its native multimodality, achieved through early fusion.[115] This means that the model is designed from the ground up to seamlessly process and understand both text and image tokens within a unified model backbone, rather than handling them as separate inputs.[115] Llama 4 Scout boasts an exceptionally large context window of 10 million tokens, significantly exceeding the capacity of previous Llama versions and many contemporary LLMs.[115] The architecture also incorporates interleaved attention layers without positional embeddings in certain parts, referred to as iRoPE.[115] The Llama 4 models were pre-trained on vast amounts of multimodal data, with approximately 40 trillion tokens for Scout and 22 trillion tokens for Maverick. The training data had a cutoff date of August 2024 and included a mix of publicly available data, licensed data, and proprietary data from Meta's products and services.[120] Llama 4 supports a range of languages, including Arabic, English, French, German, Hindi, Indonesian, Italian, Portuguese, Spanish, Tagalog, Thai, and Vietnamese.[120]

Comparison with Previous Llama Versions and Other Contemporary LLMs

Llama 4 models represent a significant step forward in the Llama family, outperforming all previous generations in terms of overall power and capabilities.[115] A particularly notable advancement is the massive 10 million token context window offered by Llama 4 Scout, which is nearly 80 times larger than the 128,000 token limit of Llama 3.[115] This expanded context window enables the model to handle much longer sequences of information, facilitating tasks like multi-document summarization and reasoning over extensive codebases.[115] Llama 4 Maverick is positioned as a

versatile and high-quality model for general assistant and chat applications, with a particular strength in image interpretation and creative writing.[116] Benchmark results indicate that Llama 4 Scout achieves better performance than other leading models in its class, including Gemma 3, Gemini 2.0 Flash-Lite, and Mistral 3.1, across a wide range of widely reported benchmarks.[115] Furthermore, Llama 4 Maverick reportedly outperforms even more powerful models like GPT-4.5 and Gemini 2.0 Pro on STEM-related benchmarks and demonstrates competitive performance with DeepSeek v3.1, a much larger model, on coding and reasoning tasks.[119] In addition to these performance improvements, Llama 4 also incorporates advancements in safety, exhibiting reduced refusal rates on debated political and social topics compared to Llama 3.[115] These comparisons highlight the continuous progress and increasing sophistication of the Llama series, positioning Llama 4 as a cutting-edge open-source offering in the rapidly evolving landscape of large language models.

Conclusion

The journey from the "Attention is All You Need" paper to the Llama 4 architecture represents a remarkable evolution in the field of Large Language Models. The introduction of the Transformer architecture, with its parallel processing capabilities and self-attention mechanism, laid a new foundation for sequence modeling, moving beyond the limitations of recurrent and convolutional networks. This breakthrough enabled the development of the first generation of Transformer-based LLMs, including GPT-1 and BERT, which showcased the potential for generative pre-training and bidirectional contextual understanding. Subsequent scaling efforts led to the emergence of even more capable models like GPT-2 and GPT-3, demonstrating emergent language capabilities and few-shot learning. While the Transformer architecture became dominant, research into alternative architectures like Mamba continues to explore avenues for improved efficiency and scalability. The Llama lineage, starting with Llama 1, has built upon the success of the Transformer, incorporating architectural refinements, scaling training data, and emphasizing open-source accessibility. The latest in this series, Llama 4, marks a significant leap forward with the adoption of a mixture of experts architecture for enhanced efficiency, native multimodality for seamless text and vision understanding, and an unprecedentedly large context window, particularly in the Scout model. This progression highlights the key trends driving LLM development: the pursuit of parallel processing, the power of self-attention, the benefits of massive scaling, the expansion into multimodal capabilities, and the increasing importance of efficiency. As research continues, the future of LLMs promises even more powerful, versatile, and accessible AI models that will undoubtedly transform numerous industries and applications.

Works cited

1. Attention Is All You Need - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Attention_Is_All_You_Need
2. “Attention Is All You Need” Explained | by Zaynab Awofeso | CodeX - Medium, accessed April 7, 2025, https://medium.com/codex/attention-is-all-you-need-explained-ebdb02c7f4d4
3. Attention is All you Need - NIPS papers, accessed April 7, 2025, https://papers.neurips.cc/paper/7181-attention-is-all-you-need.pdf
4. [1706.03762] Attention Is All You Need - arXiv, accessed April 7, 2025, https://arxiv.org/abs/1706.03762
5. Attention is All you Need - NIPS papers, accessed April 7, 2025, https://papers.nips.cc/paper/7181-attention-is-all-you-need
6. What is a Transformer Model? - IBM, accessed April 7, 2025, https://www.ibm.com/think/topics/transformer-model
7. The Evolution of Foundational Transformer Architectures in ..., accessed April 7, 2025, https://medium.com/@zbabar/the-evolution-of-foundational-transformer-architectures-in-generative-ai-483e14862b08
8. How Transformers Work: A Detailed Exploration of Transformer Architecture - DataCamp, accessed April 7, 2025, https://www.datacamp.com/tutorial/how-transformers-work
9. Innovations in AI: Transformer Technology and Free AI Tools - LLM Directory, accessed April 7, 2025, https://llmmodels.org/blog/innovations-in-ai-transformer-technology-and-free-ai-tools/
10. Unveiling the Power of Transformers: A Breakthrough in Machine Learning, accessed April 7, 2025, https://pratikbarjatya.medium.com/unveiling-the-power-of-transformers-a-breakthrough-in-machine-learning-1e9aad5a59f9
11. Transformer (deep learning architecture) - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Transformer_(deep_learning_architecture)
12. The Transformers Architecture in Detail - What's the magic behind LLMs? - AIgents.co, accessed April 7, 2025, https://aigents.co/data-science-blog/publication/the-transformers-architecture-in-detail-whats-the-magic-behind-llms
13. What is a Transformer Model? Components, Innovations & Use Cases - AI21 Labs, accessed April 7, 2025, https://www.ai21.com/knowledge/tranformer-model/
14. Attention Networks: A simple way to understand Self-Attention | by Geetansh Kalra - Medium, accessed April 7, 2025, https://medium.com/@geetkal67/attention-networks-a-simple-way-to-understand-self-attention-f5fb363c736d
15. Chapter 8 Attention and Self-Attention for NLP | Modern Approaches in Natural Language Processing, accessed April 7, 2025, https://slds-lmu.github.io/seminar_nlp_ss20/attention-and-self-attention-for-nlp.h

tml
16. arXiv:1706.03762v7 [cs.CL] 2 Aug 2023, accessed April 7, 2025, http://arxiv.org/pdf/1706.03762
17. Understanding Attention Mechanism, Self-Attention Mechanism and Multi-Head Self-Attention Mechanism | by Sapna Limbu | Medium, accessed April 7, 2025, https://medium.com/@limbusapna3/understanding-attention-mechanism-self-attention-mechanism-and-multi-head-self-attention-mechanism-94d14e937820
18. What is self-attention? | IBM, accessed April 7, 2025, https://www.ibm.com/think/topics/self-attention
19. A Comprehensive Survey on Applications of Transformers for Deep Learning Tasks - arXiv, accessed April 7, 2025, https://arxiv.org/abs/2306.07303
20. GPT-1 - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/GPT-1
21. GPT-1 | Paper Explained & PyTorch Implementation - YouTube, accessed April 7, 2025, https://www.youtube.com/watch?v=Vmqy3VP6DFk
22. Paper summary: GPT 1 — Improving Language Understanding by Generative Pre-Training, accessed April 7, 2025, https://sannaperzon.medium.com/paper-summary-gpt-1-improving-language-understanding-by-generative-pre-training-c43bd7ff242a
23. cdn.openai.com, accessed April 7, 2025, https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf
24. Improving Language Understanding by Generative Pre-Training | Papers With Code, accessed April 7, 2025, https://paperswithcode.com/paper/improving-language-understanding-by
25. Understanding the Evolution of ChatGPT: Part 1-An In-Depth Look at GPT-1 and What Inspired It | Towards Data Science, accessed April 7, 2025, https://towardsdatascience.com/understanding-the-evolution-of-gpt-part-1-an-in-depth-look-at-gpt-1-and-what-inspired-it-b7388a32e87d/
26. BERT (language model) - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/BERT_(language_model)
27. BERT Paper Explained - YouTube, accessed April 7, 2025, https://www.youtube.com/watch?v=yCj8jHnm5OQ
28. Transformer Models in Natural Language Processing - Netguru, accessed April 7, 2025, https://www.netguru.com/blog/transformer-models-in-nlp
29. arxiv.org, accessed April 7, 2025, https://arxiv.org/pdf/1810.04805
30. BERT Explained - Papers With Code, accessed April 7, 2025, https://paperswithcode.com/method/bert
31. Paper Dissected: “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding” Explained - DataScienceToday, accessed April 7, 2025, https://datasciencetoday.net/index.php/en-us/nlp/211-paper-dissected-bert-pre-training-of-deep-bidirectional-transformers-for-language-understanding-explained
32. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, accessed April 7, 2025, https://aclanthology.org/N19-1423/
33. What is the difference between BERT architecture and vanilla Transformer

architecture - Data Science Stack Exchange, accessed April 7, 2025, https://datascience.stackexchange.com/questions/86104/what-is-the-difference-between-bert-architecture-and-vanilla-transformer-archite
34. Foundation Models, Transformers, BERT and GPT | Niklas Heidloff, accessed April 7, 2025, https://heidloff.net/article/foundation-models-transformers-bert-and-gpt/
35. GPT and BERT: A Comparison of Transformer Architectures - DEV Community, accessed April 7, 2025, https://dev.to/meetkern/gpt-and-bert-a-comparison-of-transformer-architectures-2k46
36. How is BERT different from the original transformer architecture? - AI Stack Exchange, accessed April 7, 2025, https://ai.stackexchange.com/questions/23221/how-is-bert-different-from-the-original-transformer-architecture
37. When would we use a transformer encoder only (similar to BERT?), a transformer decoder only (similar to GPT?), or a transformer encoder-decoder (as proposed by Vaswani et al. in 2017)? - Reddit, accessed April 7, 2025, https://www.reddit.com/r/MLQuestions/comments/l1eiuo/when_would_we_use_a_transformer_encoder_only/
38. Uni-directional Transformer VS Bi-directional BERT - Stack Overflow, accessed April 7, 2025, https://stackoverflow.com/questions/55114128/uni-directional-transformer-vs-bi-directional-bert
39. BERT vs GPT: A Tale of Two Transformers That Revolutionized NLP | by Tavva Prudhvith, accessed April 7, 2025, https://medium.com/@prudhvithtavva/bert-vs-gpt-a-tale-of-two-transformers-that-revolutionized-nlp-11fff8e61984
40. why all the large language models are decoder-only based model? : r/LanguageTechnology, accessed April 7, 2025, https://www.reddit.com/r/LanguageTechnology/comments/11ajhat/why_all_the_large_language_models_are_decoderonly/
41. Machine Learning Glossary: Language Evaluation | Google for Developers, accessed April 7, 2025, https://developers.google.com/machine-learning/glossary/language
42. BIDIRECTIONAL LANGUAGE MODELS ARE ALSO FEW-SHOT LEARNERS - OpenReview, accessed April 7, 2025, https://openreview.net/pdf?id=wCFB37bzud4
43. [2209.14500] Bidirectional Language Models Are Also Few-shot Learners - arXiv, accessed April 7, 2025, https://arxiv.org/abs/2209.14500
44. [D] Why does the BERT paper say that standard conditional language models cannot be bidirectional? : r/MachineLearning - Reddit, accessed April 7, 2025, https://www.reddit.com/r/MachineLearning/comments/e71vyr/d_why_does_the_bert_paper_say_that_standard/
45. Large Language Models: A Frontier in AI | by Krishnarajan Arunachalam | Medium, accessed April 7, 2025,

https://medium.com/@krishnarajanarunachalam/large-language-models-a-frontier-in-ai-ecaf5308f487
46. Why can't standard conditional language models be trained left-to-right *and* right-to-left?, accessed April 7, 2025, https://stats.stackexchange.com/questions/438072/why-cant-standard-conditional-language-models-be-trained-left-to-right-and-ri
47. How do Transformers work? - Hugging Face LLM Course, accessed April 7, 2025, https://huggingface.co/learn/llm-course/chapter1/4
48. The Transformer model family - Hugging Face, accessed April 7, 2025, https://huggingface.co/docs/transformers/model_summary
49. An Overview of Different Transformer-based Language Models - The Ezra Tech Blog, accessed April 7, 2025, https://techblog.ezra.com/an-overview-of-different-transformer-based-language-models-c9d3adafead8
50. GPT-2 - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/GPT-2
51. OpenAI GPT2 - Hugging Face, accessed April 7, 2025, https://huggingface.co/docs/transformers/model_doc/gpt2
52. GPT-2 Explained | Papers With Code, accessed April 7, 2025, https://paperswithcode.com/method/gpt-2
53. cdn.openai.com, accessed April 7, 2025, https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf
54. Let's reproduce GPT-2 (124M) - YouTube, accessed April 7, 2025, https://www.youtube.com/watch?v=l8pRSuU81PU
55. Comparison Between BERT and GPT-3 Architectures | Baeldung on Computer Science, accessed April 7, 2025, https://www.baeldung.com/cs/bert-vs-gpt-3-architecture
56. GPT-3 - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/GPT-3
57. Explaining GPT-3. Architecture and Working | by Abhi Sai | Medium, accessed April 7, 2025, https://medium.com/@tsaiabhi.cool/explaining-gpt-3-architecture-and-working-d0219c79202c
58. A Beginner's Guide to GPT-3 - DataCamp, accessed April 7, 2025, https://www.datacamp.com/blog/a-beginners-guide-to-gpt-3
59. [2005.14165] Language Models are Few-Shot Learners - arXiv, accessed April 7, 2025, https://arxiv.org/abs/2005.14165
60. arxiv.org, accessed April 7, 2025, https://arxiv.org/pdf/2005.14165
61. Language Models are Few-Shot Learners - NIPS papers, accessed April 7, 2025, https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
62. GPT-3 Explained | Papers With Code, accessed April 7, 2025, https://paperswithcode.com/method/gpt-3
63. Language Model History — Before and After Transformer: The AI Revolution - Medium, accessed April 7, 2025, https://medium.com/@kirudang/language-model-history-before-and-after-transf

ormer-the-ai-revolution-bedc7948a130
64. Unleashing the Power of BERT: How the Transformer Model Revolutionized NLP - Arize AI, accessed April 7, 2025, https://arize.com/blog-course/unleashing-bert-transformer-model-nlp/
65. Recurrent neural network - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Recurrent_neural_network
66. Evolution of Neural Networks to Large Language Models - Labellerr, accessed April 7, 2025, https://www.labellerr.com/blog/evolution-of-neural-networks-to-large-language-models/
67. Language model - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Language_model
68. (PDF) Recurrent neural network based language model - ResearchGate, accessed April 7, 2025, https://www.researchgate.net/publication/221489926_Recurrent_neural_network_based_language_model
69. 1. History Of Large Language Models | From 1940 To 2023 - AI Researcher, accessed April 7, 2025, https://ai-researchstudies.com/history-of-large-language-models-from-1940-to-2023/
70. The Epic History of LLMs : Journey from RNNs to ChatGPT | by Sachinsoni - Medium, accessed April 7, 2025, https://medium.com/@sachinsoni600517/the-epic-history-of-llms-journey-from-rnns-to-chatgpt-8b6c72b40f09
71. Language Models: Past, Present, and Future - Communications of the ACM, accessed April 7, 2025, https://cacm.acm.org/research/language-models/
72. A brief history of language models | Towards Data Science, accessed April 7, 2025, https://towardsdatascience.com/a-brief-history-of-language-models-d9e4620e025b/
73. Differences Between Bidirectional and Unidirectional LSTM | Baeldung on Computer Science, accessed April 7, 2025, https://www.baeldung.com/cs/bidirectional-vs-unidirectional-lstm
74. The Bidirectional Language Model. Easy trick to include both left and... | by Motoki Wu | Medium, accessed April 7, 2025, https://medium.com/@plusepsilon/the-bidirectional-language-model-1f3961d1fb27
75. Chapter 5 Convolutional neural networks and their applications in NLP | Modern Approaches in Natural Language Processing, accessed April 7, 2025, https://slds-lmu.github.io/seminar_nlp_ss20/convolutional-neural-networks-and-their-applications-in-nlp.html
76. LLaMA Explained!. Llama is one of the leading state of... | by Pranjal Khadka - Towards AI, accessed April 7, 2025, https://pub.towardsai.net/llama-explained-a70e71e706e9
77. The Evolution of Llama: From Llama 1 to Llama 3.1 - Artificial Intelligence, accessed

April 7, 2025, https://zaai.ai/the-evolution-of-llama-from-llama-1-to-llama-3-1/
78. LLaMA Explained | Papers With Code, accessed April 7, 2025, https://paperswithcode.com/method/llama
79. Mastering LLaMA: A Deep Dive into Meta AI's Revolutionary Model | by Ebad Sayed, accessed April 7, 2025, https://medium.com/@sayedebad.777/mastering-llama-a-deep-dive-into-meta-ais-revolutionary-model-07886186480b
80. Introducing LLaMA: A foundational, 65-billion-parameter large language model - Meta AI, accessed April 7, 2025, https://ai.meta.com/blog/large-language-model-llama-meta-ai/
81. [2302.13971] LLaMA: Open and Efficient Foundation Language Models - arXiv, accessed April 7, 2025, https://arxiv.org/abs/2302.13971
82. LLaMA Model Architecture: A Comprehensive Guide - BytePlus, accessed April 7, 2025, https://www.byteplus.com/en/topic/406459
83. A Complete Beginner's Guide to Llama 2 | Build Generative AI Applications With SingleStoreDB, accessed April 7, 2025, https://www.singlestore.com/blog/a-complete-beginners-guide-to-llama2/
84. Llama (language model) - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Llama_(language_model)
85. Introduction to Llama2 : Part-1 Architectural Analysis | by Utsavtiwari - Medium, accessed April 7, 2025, https://medium.com/@utsavtiwari9936/introduction-to-llama2-part-1-architectural-analysis-3e335e7b1104
86. Meta Llama 2, accessed April 7, 2025, https://www.llama.com/llama2/
87. Llama2 - Hugging Face, accessed April 7, 2025, https://huggingface.co/docs/transformers/model_doc/llama2
88. aju22/LLaMA2: This repository contains an implementation of the LLaMA 2 (Large Language Model Meta AI) model, a Generative Pretrained Transformer (GPT) variant. The implementation focuses on the model architecture and the inference process. The code is restructured and heavily commented to facilitate easy understanding of the key parts of the architecture - GitHub, accessed April 7, 2025, https://github.com/aju22/LLaMA2
89. AI Breakdown or: Takeaways From the 78-page Llama-2 Paper | Deepgram, accessed April 7, 2025, https://deepgram.com/learn/llama-2-paper-explained
90. Understanding LLaMA-2 Architecture & its Ginormous Impact on GenAI | by Kunal Sawarkar | Towards Generative AI | Medium, accessed April 7, 2025, https://medium.com/towards-generative-ai/understanding-llama-2-architecture-its-ginormous-impact-on-genai-e278cb81bd5c
91. [2307.09288] Llama 2: Open Foundation and Fine-Tuned Chat Models - arXiv, accessed April 7, 2025, https://arxiv.org/abs/2307.09288
92. Convolutional Neural Networks (CNNs): A 2025 Deep Dive - viso.ai, accessed April 7, 2025, https://viso.ai/deep-learning/convolutional-neural-networks/
93. History of CNN & its impact in the field of Artificial Intelligence | by Daksh Bhatnagar, accessed April 7, 2025, https://medium.com/accredian/history-of-cnn-its-impact-in-the-field-of-artificial

-intelligence-2b1efb7d99e5
94. Convolutional neural network - Wikipedia, accessed April 7, 2025, https://en.wikipedia.org/wiki/Convolutional_neural_network
95. The history of convolutional neural networks. : r/ArtificialInteligence - Reddit, accessed April 7, 2025, https://www.reddit.com/r/ArtificialInteligence/comments/1cjhfsf/the_history_of_convolutional_neural_networks/
96. The Evolution of Language Models: A Journey Through Time | by Adria Cabello | Medium, accessed April 7, 2025, https://medium.com/@adria.cabello/the-evolution-of-language-models-a-journey-through-time-3179f72ae7eb
97. The History of Convolutional Neural Networks - Glass Box, accessed April 7, 2025, https://glassboxmedicine.com/2019/04/13/a-short-history-of-convolutional-neural-networks/
98. Convolutional Neural Networks: 1998-2023 Overview - SuperAnnotate, accessed April 7, 2025, https://www.superannotate.com/blog/guide-to-convolutional-neural-networks
99. Alternatives to Transformer based Architectures | by Digvijay Y - Medium, accessed April 7, 2025, https://medium.com/@digvijay.qi/alternatives-to-transformer-based-architectures-3f41faeaacab
100. Mamba (Transformer Alternative): The Future of LLMs and ChatGPT? - Lazy Programmer, accessed April 7, 2025, https://lazyprogrammer.me/mamba-transformer-alternative-the-future-of-llms-and-chatgpt/
101. Ask HN: Is anybody building an alternative transformer? - Hacker News, accessed April 7, 2025, https://news.ycombinator.com/item?id=43052427
102. [Discussion] Promising alternatives to the standard transformer? : r/MachineLearning - Reddit, accessed April 7, 2025, https://www.reddit.com/r/MachineLearning/comments/164n8iz/discussion_promising_alternatives_to_the_standard/
103. [D] Which architecture could substitute the transformer? : r/MachineLearning - Reddit, accessed April 7, 2025, https://www.reddit.com/r/MachineLearning/comments/18apkw6/d_which_architecture_could_substitute_the/
104. Transformer alternatives in 2024 - Nebius AI, accessed April 7, 2025, https://nebius.com/blog/posts/model-pre-training/transformer-alternatives-2024
105. Transformer alternatives in 2024 - Medium, accessed April 7, 2025, https://medium.com/nebius/transformer-alternatives-in-2024-06cd3d91d42b
106. MoE vs Dense vs Hybrid LLM architectures | hybridMoe – Weights & Biases - Wandb, accessed April 7, 2025, https://wandb.ai/zaiinn440/hybridMoe/reports/MoE-vs-Dense-vs-Hybrid-LLM-architectures--VmlldzO3NzYwNzAw
107. Llama 2 Explained: Training, Performance and Results, accessed April 7, 2025, https://alexandrabarr.beehiiv.com/p/llama-2

108. Fine-Tuning Llama 3 with LoRA: Step-by-Step Guide - neptune.ai, accessed April 7, 2025, https://neptune.ai/blog/fine-tuning-llama-3-with-lora
109. llama3/MODEL_CARD.md at main · meta-llama/llama3 - GitHub, accessed April 7, 2025, https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md
110. Introducing Meta Llama 3: The most capable openly available LLM to date, accessed April 7, 2025, https://ai.meta.com/blog/meta-llama-3/
111. Llama-3, A Deep Dive | The Critical Section, accessed April 7, 2025, https://aceofgreens.github.io/llama_3.html
112. Build Your Own Llama 3 Architecture from Scratch Using PyTorch | by Milan Tamang, accessed April 7, 2025, https://pub.towardsai.net/build-your-own-llama-3-architecture-from-scratch-using-pytorch-2ce1ecaa901c
113. Llama 3 Guide: Everything You Need to Know About Meta's New Model and Its Data, accessed April 7, 2025, https://kili-technology.com/large-language-models-llms/llama-3-guide-everything-you-need-to-know-about-meta-s-new-model-and-its-data
114. Training Techniques for Llama 3 Model | Restackio, accessed April 7, 2025, https://www.restack.io/p/llama-3-answer-training-techniques-cat-ai
115. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation - Meta AI, accessed April 7, 2025, https://ai.meta.com/blog/llama-4-multimodal-intelligence/
116. Meta Unveils Llama 4 AI Series Featuring New Expert-Based Architecture - TechRepublic, accessed April 7, 2025, https://www.techrepublic.com/article/news-meta-llama-4-models/
117. Introducing the Llama 4 herd in Azure AI Foundry and Azure Databricks | Microsoft Azure Blog, accessed April 7, 2025, https://azure.microsoft.com/en-us/blog/introducing-the-llama-4-herd-in-azure-ai-foundry-and-azure-databricks/
118. Meta's Llama 4 is now available on Workers AI - The Cloudflare Blog, accessed April 7, 2025, https://blog.cloudflare.com/meta-llama-4-is-now-available-on-workers-ai/
119. Meta's New Llama 4's MoE Architecture Makes AI Faster & Cheaper | by Tahir - Medium, accessed April 7, 2025, https://medium.com/@tahirbalarabe2/metas-new-llama-4-s-moe-architecture-makes-ai-faster-cheaper-635339e51e10
120. llama-models/models/llama4/MODEL_CARD.md at main - GitHub, accessed April 7, 2025, https://github.com/meta-llama/llama-models/blob/main/models/llama4/MODEL_CARD.md
121. meta-llama/Llama-4-Scout-17B-16E-Original - Hugging Face, accessed April 7, 2025, https://huggingface.co/meta-llama/Llama-4-Scout-17B-16E-Original
122. Welcome Llama 4 Maverick & Scout on Hugging Face, accessed April 7, 2025, https://huggingface.co/blog/llama4-release