Review Article

Journal of Autonomous Vehicles, Drones, and Intelligent Mobility

A Review of Multimodal Vision–Language Models: Foundations, Applications, and Future Directions

Gurpreet Singh

Independent Researcher

*Corresponding author: Gurpreet Singh, Independent Researcher.

Submitted: 11 November 2025 **Accepted:** 17 November 2025 **Published:** 25 November 2025

https://doi.org/10.63620/MKJAVDIM.2025.

Citation: *Gurpreet Singh. (2025). A Review of Multimodal Vision–Language Models: Foundations, Applications, and Future Directions. J Aut Veh Dro and Int Mob, 1(2), 01-12.*

Abstract

Large Language Models (LLMs) have rapidly become a central focus in both research and practical applications, owing to their remarkable ability to understand and generate text with a level of fluency comparable to human communication. Recently, these models have evolved into multi- modal large language models (MM-LLMs), extending their capabilities beyond text to include images, audio, and video. This advancement has enabled a wide array of applications, including text-to-video synthesis, image captioning, and text-to-speech systems. MM-LLMs are developed either by augmenting existing LLMs with multi-modal functionality or by designing multi-modal architectures from the ground up. This paper presents a comprehensive review of the current landscape of LLMs with multi-modal capabilities, highlighting both foundational and cutting-edge MM-LLMs. It traces the historical development of LLMs, emphasizing the transformative impact of transformer-based architectures such as OpenAI's GPT series and Google's BERT, as well as the role of attention mechanisms in improving model performance. The review also examines key strategies for adapting pre-trained models to specific tasks, including fine-tuning and prompt engineering. Ethical challenges, including data bias and the potential for misuse, are discussed to stress the importance of responsible AI deployment. Finally, we explore the im- plications of open-source versus proprietary models for advancing research in this field. By synthesizing these insights, this paper underscores the significant potential of MM-LLMs to reshape diverse applications across multiple domains.

Keywords: Large Language Models (LLMs), Multi-Modal Large Language Models (MM-LLMs), Transformer Architecture, GPT, BERT, Attention Mech- anism, Fine-Tuning, Prompt Engineering, Text-to-Video Generation, Image Captioning.

Introduction

Large Language Models (LLMs) have emerged as one of the most prominent topics in contemporary artificial intelligence (AI) research, with growing interest not only in academic circles but also in the broader public. Their visibility has been amplified by the release of ChatGPT in 2022, which showcased the potential of LLMs to generate coherent, human-like text. By LLMs, we refer specifically to language models built on the Transformer architecture, such as OpenAI's Generative Pre-trained Transformers (GPT), which began with GPT-1 in 2018 [1-5]. The increasing prominence of LLMs stems from their demonstrated versatility across a wide spectrum of tasks, including text summarization, text- to-image and text-to-video generation, conversational search, machine translation, and broader generative AI (GenAI) applications. A systematic review of over 1,300 related publications underscores their central role in advancing GenAI [6-10].

Beyond OpenAI's GPT series, other notable proprietary LLMs attracting pub- lic and research attention include Google's Gemini/BARD and Anthropic's Claude [11-15]. At the same time, several high-profile open-source models, such as Meta's LLaMA, Google's PaLM, and Falcon from the UAE's Technology Innovation Institute, have been introduced to the community.

The release or update of any LLM can generate significant interest both within academia and in the media, making it challenging to track developments, compare model capabilities, and identify their specific applications [16-18].

This review focuses on LLMs with particular attention to visual and multi- modal capabilities (MM-LLMs), examining their architectures, optimization strategies, and application-specific adaptation. While prior work provides a concise overview of LLMs covering their history, architecture, training strategies, applications, and challenges, it does not extensively address models capable of processing and generating multiple modalities, such as text, images, audio, and video. Our work complements this literature by analyzing the technical aspects of MM-LLMs, including open-source versus proprietary models, computational considerations, and strategies for efficient fine-tuning. We also explore practical aspects, such as which architectural or training components are most relevant for reducing cost and improving model performance, as well as the evaluation techniques commonly used to assess LLM quality.

Ethical considerations are increasingly central to discussions around LLMs. Concerns highlighted in the literature include potential data biases, environmental and energy costs, and the concentration of powerful models within a few large technology companies. This review examines these issues in the context of MM-LLMs, particularly open-source implementations, and evaluates how they can be deployed responsibly in practical multimedia applications.

What is a Language Model?

The Evolution of Language Models

Language Models (LMs) have long been a cornerstone of Natural Language Pro- cessing (NLP), forming the foundation for a wide range of text-based applications. Traditionally, LMs relied on statistical methods, where models were trained on large text corpora to predict the next word in a sequence. By analyzing patterns, frequency, and context in text, these models sought to capture the structure and nuances of human language [19, 20]
The journey from early LMs to today's Large Language Models (LLMs) reflects significant advances in NLP. Initially, NLP systems were rule-based, designed for applications like machine translation and speech recognition. These approaches gradually gave way to statistical methods, such as Hidden Markov Models and N-gram models. While effective at capturing short-term word dependencies, these models struggled with long-range context and semantic understanding. Neural Networks (NNs), first conceptualized in the 1950s, were not widely applied to NLP until computational resources became sufficient to handle their demands [21].

A major breakthrough came in the 2010s with the advent of word embedding techniques, notably Word2Vec and GloVe [22]. Word embeddings represent words as continuous vectors within a semantic space, enabling models to capture relationships and similarities between words. This innovation laid the groundwork for the resurgence of deep learning approaches in NLP.

The next leap forward involved Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, which allowed models to process sequences of words in a more context-aware manner [23]. Unlike N- gram models, which were limited to nearby words, RNNs and LSTMs could theoretically capture dependencies across entire sequences. Nevertheless, these models had their limitations: as sequences became longer, it became increasingly difficult to retain relevant context, and parallelizing computations was not feasible, creating bottlenecks in training [24].

The field underwent a transformative shift in 2017 with the introduction of the Transformer architecture [25]. One of the key innovations of Transformers was the attention mechanism, which allows models to evaluate the importance of different parts of an input sequence in parallel. This approach resolved the long-range dependency problem inherent in RNNs and LSTMs, enabling models to capture relationships across entire sequences more effectively [26].

Building on the Transformer, 2018 saw the release of two landmark models. Google introduced Bidirectional Encoder Representations from Transformers (BERT), while OpenAI launched its first Generative Pre-trained Transformer (GPT) [27]. Together with the availability of massive text datasets and improved computational power, these models established the foundation for modern LLMs [28].

Although these models were already impressive, widespread public and research interest surged with the launch of OpenAI's ChatGPT in November 2022. ChatGPT demonstrated the practical ability of LLMs to engage in natural, conversational interactions, summarize documents, and support various generative AI tasks [29].

Attention Mechanisms

The Transformer architecture represents a paradigm shift in NLP, relying solely on attention mechanisms to process and understand text sequences [30]. Among the most widely used are Self-Attention and Multi-Head Attention, which form the backbone of modern LLMs.

Self-Attention enables a model to weigh the importance of different positions within a single sequence, generating a context-aware representation. The input is decomposed into linear query, key, and value vectors, allowing the model to focus on the most relevant parts of the text.

Multi-Head Attention (MHA) extends this idea by computing multiple self- attention operations in parallel, with each "head" attending to different aspects of the input sequence. While MHA provides richer contextual understanding, it can be computationally demanding and may strain memory resources.

To address this, Multi-Query Attention (MQA) was proposed. MQA reduces memory usage by sharing a single key and value across multiple query heads, which allows for larger batch sizes and faster computation. The trade-off is a potential reduction in attention detail, as fewer key-value pairs may overlook subtle aspects of the input.

Grouped-Query Attention (GQA) offers a middle ground between MHA and MQA. In GQA, queries are grouped and assigned to corresponding key-value pairs, preserving more detail than MQA while being faster than MHA. This design allows models to process longer sequences efficiently without significant loss of context or performance [31].

Proprietary vs. Open Source LLMs

In 2023, research and development in the field of large language models continued at a rapid pace, with major technology companies like OpenAI and Google striving to create the most advanced models. Historically, it was assumed that larger LLMs would provide a competitive edge. However, building such models required significant financial investment, often ranging from €1 million to over €100 million, due to the immense datasets and GPU resources needed. The release of Meta's open-source LLaMA marked a turning point, reflecting the belief that freely available models could stimulate innovation, enhance safety, and encourage responsible AI practices [32]. Today, several open-source LLMs, such as Meta's LLaMA-2 and Google's PaLM 2, can be accessed without charge, whereas proprietary models like OpenAI's GPT or Google's BARD typically impose usage-based fees for enterprise access [33, 34].

Even Google has acknowledged the inherent limitations of proprietary mod- els, recognizing that open-source alternatives could quickly match or surpass their own LLMs. Open-source communities have already tackled challenges that proprietary developers had struggled with, accelerating innovation outside of corporate constraints [35, 36]. Open-source LLMs offer clear advantages for researchers and entrepreneurs. They are cost-effective in the long term and pro- vide transparency into the model architecture, training data, and methodologies, which facilitates auditing and ensures compliance with ethical and legal stan- dards. This is especially relevant in light of regulatory frameworks such as the European Union AI Act, expected in 2025, which will require openness regarding model training data for commercial deployment in the EU [37]. Open-source LLMs also give researchers complete control over the data used for fine-tuning, minimizing the risk of sensitive information leaks. Additionally, optimizing open-source models can improve computational efficiency, reduce latency, and enhance performance for specific applications.

Despite these benefits, open-source LLMs also carry limitations. They often lack formal service agreements, leaving developers without guaranteed support or ongoing updates. The pace of innovation in open-source communities can be unpredictable, while proprietary models may remain more stable and reliable in certain contexts. Furthermore, not all open-source models are entirely unre- stricted. For example, Meta's LLaMA-2 enforces usage conditions through its acceptable use policy [38].

Key Large Language Models

This section provides an overview of prominent LLMs, focusing primarily on models designed for text generation. Some of these models have been adapted to incorporate multi-modal capabilities post hoc.

GPT

The GPT family, developed by OpenAI, represents a lineage of LLMs beginning with GPT-1. GPT stands for Generative Pre-trained Transformer, and the initial model used a 12-layer decoder-only transformer with masked self-attention heads, trained on a large, diverse text corpus. GPT-1 demonstrated improvements in NLP benchmarks across several datasets [39].

Subsequent iterations, including GPT-2 and GPT-3, scaled up both in model size and training data. GPT-2, with 1.5 billion parameters, showed that language models could achieve strong performance in tasks such as text comprehension and summarization without supervision. GPT-3 expanded this scale dramatically, reaching 175 billion parameters, highlighting the advantages of larger model ca- pacity [40]. GPT-3.5, the engine behind ChatGPT, brought these capabilities to the public in 2022, popularizing generative AI applications. GPT-4 further enhanced this family by incorporating multi-modal capabilities, accepting both text and images as input while producing text or image outputs. GPT-4 has demonstrated substantial improvements on NLP benchmarks, including perfor- mance on the bar exam at the 90th percentile, compared with GPT-3.5, which scored in the bottom decile. Nonetheless, GPT-4 shares limitations with its predecessors, including hallucinations, limited context windows, and an inability to learn incrementally. It is trained on a combination of public and licensed datasets and fine-tuned using Reinforcement Learning from Human Feedback (RLHF) [41].

Claude

Anthropic's Claude models, first released as Claude 1 in March 2024, are designed for NLP tasks such as summarization, question answering, and code generation. Claude is notable for its emphasis on safety and controlled outputs, aiming to reduce harmful or biased responses [42]. Claude 3, introduced in 2023, comes in three variants—Opus, Sonnet, and Haiku—and includes multi-modal functionality, allowing it to process visual inputs. It reportedly features an extremely large context window of up to 200,000 tokens, enabling the model to consider very long input sequences in a single pass. Claude 3 is trained on a mix of public, private, and synthetic datasets, with public data sourced up to August 2023. While the architecture details remain largely undisclosed, Anthropic claims that Claude 3 matches or exceeds the performance of other leading LLMs.

Gemini

Google's Gemini family, introduced in 2023, represents a set of models capable of handling text, images, audio, and video. Gemini has achieved state-of-the-art results across multiple benchmarks, notably the Gemini Ultra variant, which scored 64% on the MMMU benchmark involving multi-disciplinary image and text tasks, surpassing previous models by over 5 percentage points [43]. The Gemini family includes Ultra, Pro, and Nano versions, which vary in size. All models employ the Transformer architecture with Multi-Query Attention (MQA) and can process inputs up to 32,000 tokens. A key distinction between Gemini and GPT-4 is Gemini's ability to generate images as outputs, in addition to text.

LLaMA

Meta AI's LLaMA series is an open-source collection of LLMs designed to democratize access to large-scale language models for research and development. The models range from 7 billion to 65 billion parameters and are optimized for inference speed, even allowing operation on a single GPU [44]. Contrary to the "bigger is better" assumption, research by Hoffman et al. suggests that smaller models trained on larger datasets can outperform larger models given the same computational budget. LLaMA models were trained exclusively on publicly available datasets, avoiding proprietary data, with the goal of maximizing performance per computational cost. LLaMA-13B has been shown to outperform GPT-3 on multiple benchmarks, while LLaMA-65B remains competitive with larger models such as Chinchilla or PaLM-540B [44-50].

The architecture of LLaMA is grounded in the original Transformer framework, with enhancements such as pre-normalization (as in GPT-3), the SwiGLU activation function, and Rotary Positional Embeddings (RoPE) for efficient positional encoding. These modifications improve stability, reduce compu- tational load, and preserve important positional information. Benchmarking against other LLMs in tasks like zero-shot and few-shot learning, common- sense reasoning, reading comprehension, and code generation demonstrates that LLaMA achieves competitive or superior performance. Additionally, small-scale fine-tuning improves results on multi-task benchmarks like MMLU. Evaluation of LLaMA-65B also considers fairness and safety metrics, including truthfulness, bias, and toxicity, using datasets such as RealToxicityPrompts, CrowS-Pairs, WinoGender, and TruthfulQA [51, 52]

LLaMA-2 and LLaMA-2 Chat

In July 2023, Meta AI released LLaMA-2 along with LLaMA-2 Chat, representing a substantial update to the original LLaMA series. These models range in size from 7 billion to 70 billion parameters and incorporate several key improvements. One significant enhancement was the expansion of the pre-training corpus by 40%, alongside doubling the context window from 2,048 to 4,096 tokens. A major distinction between LLaMA-2 (and its Chat variant) and the original LLaMA is the adoption of Reinforcement Learning from Human Feedback (RLHF) during fine-tuning, a method further discussed in Section 6.1.

While continuing to rely on publicly available datasets for training, LLaMA-2 integrates additional data and enhanced safety measures to reduce the risk of generating unsafe outputs. Unlike its predecessor, which was limited to a non-commercial open-source license, LLaMA-2 introduces a commercial license to promote collaborations and broaden potential applications. Meta AI has also released the model weights and initial code to support researchers and developers in extending or customizing these models. Architecturally, LLaMA-2 follows the same transformer-based framework as LLaMA but integrates the Grouped-Query Attention (GQA) mechanism to improve efficiency and processing capability [53-56].

Benchmark evaluations show that LLaMA-2 outperforms most other open- source LLMs across a range of tasks, with the exception of coding-focused challenges. When compared to proprietary models like GPT-4 or PaLM-2, its performance is generally lower but aligns closely with GPT-3.5 and PaLM in overall outcomes [57-62].

MedAlpaca

MedAlpaca, introduced in October 2023, represents a specialized adaptation of LLaMA models for biomedical applications. Developed using open-source biomedical datasets, its primary aim is to provide on-premises deployment capabilities to protect sensitive patient data, a crucial requirement in healthcare settings. The model employs Low-Rank Adaptation (LoRA) and Supervised Fine-Tuning (SFT) techniques, both of which are detailed in Section 6.1 [63-66]. MedAlpaca's performance was evaluated using the United States Medical Licensing Examination (USMLE), a standard benchmark for medical competence. Notably, MedAlpaca 13B demonstrated improved performance over the base LLaMA 13B model in zero-shot evaluation, achieving 47.3%, 47.7%, and 60.2% on Steps 1, 2, and 3, respectively. However, when LoRA fine-tuning was applied, performance decreased significantly to 25.0%, 25.5%, and 25.5%, suggesting that while LoRA is computationally efficient, it may not be the optimal choice compared to SFT for certain biomedical tasks.

Mistral 7B

Mistral 7B is a 7-billion parameter language model designed to achieve high efficiency and competitive performance despite its relatively smaller size. Ac- cording to the developers, chat models built on Mistral 7B outperform the LLaMA-2 13B Chat model. The model leverages the Grouped-Query Attention (GQA) mechanism, similar to LLaMA-2, along with Sliding Window Attention (SWA), which originates from the Longformer architecture [67]. SWA enables more efficient handling of long sequences, with stacked transformers functioning similarly to convolutional layers in CNNs, improving both performance and computational efficiency.

Falcon-7B and Falcon-40B

In May 2023, the Technology Innovation Institute (TII) in Abu Dhabi launched the Falcon series, including Falcon-7B, Falcon-40B, and their instruction-tuned counterparts: Falcon-7B-Instruct and Falcon-40B-Instruct. Released under the Apache 2.0 license, these models support unrestricted commercial use, encourag- ing widespread adoption and fine-tuning for various applications. Falcon-Instruct variants are specifically optimized for conversational and instruction-following tasks [68-72].

TII also provided a high-quality pre-training dataset, RefinedWeb, which includes 600 billion tokens derived from CommonCrawl. The dataset underwent large-scale deduplication and strict filtering to ensure quality. Architecturally, Falcon models are transformer-based, utilizing MQA for memory-efficient processing and RoPE for positional encoding. Additionally, Falcon employs Flash Attention, which optimizes speed and memory usage through tiling and recom- putation strategies, enabling faster training and longer context windows. Unlike LLaMA, Falcon does not implement the SwiGLU activation function, prioritizing memory efficiency over incremental performance gains [73-75].

Falcon models have been trained on 1.5 trillion tokens, and the curated pre-training data is considered a significant factor in their performance. The emphasis on data quality demonstrates the importance of high-fidelity datasets in building effective LLMs [76-80].

Falcon-180B

In September 2023, TII expanded the Falcon series with Falcon-180B, a 180- billion parameter model trained on 3.5 trillion tokens from the RefinedWeb dataset—more than double the amount used for previous Falcon models. A variant, Falcon-

180B-chat, was fine-tuned for instruction and conversational tasks. This model achieves competitive results relative to leading models such as GPT-4, GPT-3.5, and PaLM 2-Large [81-85].

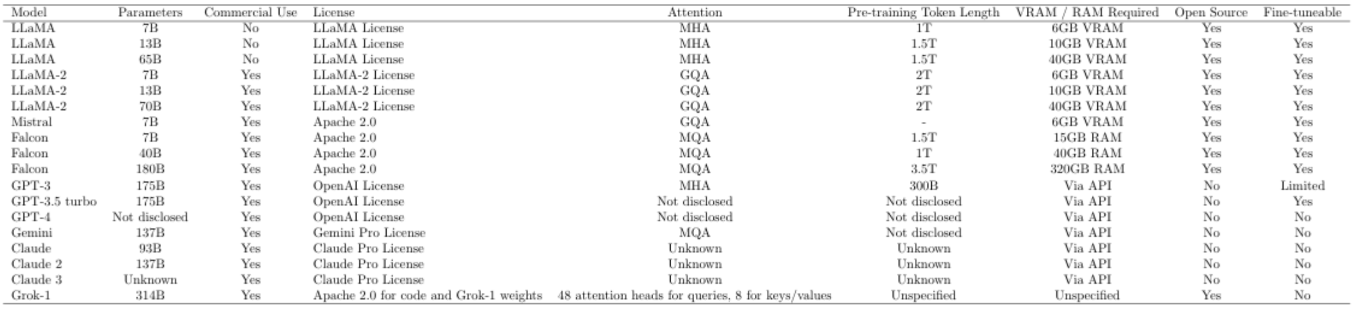
However, the model's large scale comes with substantial hardware require- ments: Falcon-180B demands at least 320GB of memory for optimal operation, compared to 40GB for Falcon-40B and 15GB for Falcon-7B. This significant memory requirement reduces accessibility for researchers with limited hardware, which is otherwise an advantage of open-source LLMs [86].

Benchmark evaluations for the Falcon series include common-sense reasoning tasks such as HellaSwag, Winogrande, AI2 Reasoning Challenge (ARC), MMLU, and Open BookQA, along with PIQA and BoolQ. These tasks are discussed further in Section 7 [87, 88].

Grok-1

Grok-1, released in March 2024 by xAI under OpenAI, is a cutting-edge LLM with 314 billion parameters. Its architecture is autoregressive and Transformer-based, featuring a mixture of eight experts. On the HumanEval coding benchmark, Grok-1 achieves 63.2% and scores 73% on MMLU. While it does not outperform models trained on larger datasets, such as GPT-4 or Claude 2, it exceeds the performance of other models trained on comparable dataset sizes.

Table 1: A Comparative Summary of the Reviewed LLMs



Model	Parameters	Commercial Use	License	Attention	Pre-training Token Length	VRAM / RAM Required	Open Source	Fine-tuneable
LLaMA	7B	No	LLaMA License	MHA	1T	6GB VRAM	Yes	Yes
LLaMA	13B	No	LLaMA License	MHA	1.5T	10GB VRAM	Yes	Yes
LLaMA	65B	No	LLaMA License	MHA	1.5T	40GB VRAM	Yes	Yes
LLaMA-2	7B	Yes	LLaMA-2 License	GQA	2T	6GB VRAM	Yes	Yes
LLaMA-2	13B	Yes	LLaMA-2 License	GQA	2T	10GB VRAM	Yes	Yes
LLaMA-2	70B	Yes	LLaMA-2 License	GQA	2T	40GB VRAM	Yes	Yes
Mistral	7B	Yes	Apache 2.0	GQA	-	6GB VRAM	Yes	Yes
Falcon	7B	Yes	Apache 2.0	MQA	1.5T	15GB RAM	Yes	Yes
Falcon	40B	Yes	Apache 2.0	MQA	1T	40GB RAM	Yes	Yes
Falcon	180B	Yes	Apache 2.0	MQA	3.5T	320GB RAM	Yes	Yes
GPT-3	175B	Yes	OpenAI License	MHA	300B	Via API	No	Limited
GPT-3.5 turbo	175B	Yes	OpenAI License	Not disclosed	Not disclosed	Via API	No	Yes
GPT-4	Not disclosed	Yes	OpenAI License	Not disclosed	Not disclosed	Via API	No	No
Gemini	137B	Yes	Gemini Pro License	MQA	Not disclosed	Via API	No	No
Claude	93B	Yes	Claude Pro License	Unknown	Unknown	Via API	No	No
Claude 2	137B	Yes	Claude Pro License	Unknown	Unknown	Via API	No	No
Claude 3	Unknown	Yes	Claude Pro License	Unknown	Unknown	Via API	No	No
Grok-1	314B	Yes	Apache 2.0 for code and Grok-1 weights	48 attention heads for queries, 8 for keys/values	Unspecified	Unspecified	Yes	No

Vision Models and Multi-Modal Large Lan- guage Models

Up to this point, we have reviewed prominent Large Language Models (LLMs) that originated primarily in the text domain, some of which later incorporated multi-modal functionalities. In this section, we shift focus to models created specifically to bridge vision and language. Vision models are engineered to produce joint representations of images and text, enabling tighter integration of visual and linguistic information than retrofitted multi-modal variants. These models underpin tasks such as automatic image captioning, cross-modal retrieval, and text-driven image generation.

Vision Models

BLIP-2

Introduced by Salesforce in 2023, BLIP-2 proposes a two-stage pretraining approach that leverages strong off-the-shelf image encoders and language models to strengthen vision–language alignment [89-92]. A central innovation is the Querying Transformer (Q-Former), which functions as an adapter between the frozen image encoder and the language model. During the first pretraining stage, the Q-Former learns to extract a compact set of visual tokens that are most relevant to textual descriptions. In the second stage, those learned visual queries feed into a frozen language model, effectively acting as soft visual prompts that guide generation. By freezing large foundation components and training only the bridging layer, BLIP-2 achieves efficient cross-modal integration while capitalizing on the representational power of pretrained vision and language backbones.

Vision Transformer (ViT)

The Vision Transformer (ViT) demonstrated that transformer architectures, originally conceived for sequential text, can be repurposed effectively for image tasks [93]. ViT divides each image into a grid of patches, flattens these patches into a sequence, and feeds that sequence into a standard transformer encoder. When pretrained on large datasets, ViT models can match or exceed the per- formance of many convolutional neural networks while simplifying architectural choices and enabling straightforward scaling. Unlike many NLP transformers, ViT commonly routes the encoder output into an MLP classification head rather than an attention-based decoder. The concept of patch embedding was pivotal in establishing the transformer's ability to generalize to visual data.

Contrastive Language–Image Pretraining (CLIP)

CLIP (Contrastive Language–Image Pretraining) quickly became a foundational method for building multi-modal systems [94]. Trained contrastively on hundreds of millions of image–text pairs, CLIP jointly learns an image encoder and a text encoder so that corresponding images and captions are close in a shared embedding space. This training paradigm confers strong zero-shot classification abilities—allowing CLIP to map natural language labels to images without task-specific supervised examples. However, CLIP's large, web-scale training corpus also exposes it to dataset bias and undesirable correlations; early analyses found problematic misclassifications that disproportionately affected certain de- mographic groups. Later advancements such as RA-CLIP (Retrieval-Augmented CLIP) sought to mitigate data and retrieval limitations by augmenting the training process with an external retrieval mechanism, yielding substantial gains in zero-shot classification performance [95].

Early Approaches to Multi-Modal Processing

Initial attempts to combine vision and language followed an encoder–decoder template inspired by machine translation: a CNN encoder would extract vi- sual features, and an RNN decoder would produce captions from that fixed representation [96]. While intuitive, these models often struggled to capture fine-grained semantics and required costly recurrent computation. Later work showed that web-scale image–text pairs could enable zero-shot annotation and more flexible multimodal behavior, shifting emphasis away from tightly coupled encoder–decoder pipelines toward contrastive and transformer-based approaches [97].

Multi-Modal Large Language Models (MM-LLMs)

Modern image-grounded MM-LLMs typically consist of three components: a vision encoder that produces visual embeddings, a language model that handles text, and an alignment or cross-modal module that connects the two. These systems aim to provide unified multimodal reasoning and generation rather than simply appending visual inputs to a text model. Below we review representative MM-LLMs and how they achieve cross-modal competence.

LLaVA (Large Language and Vision Assistant)

LLaVA couples an image encoder (often CLIP-based) with a strong LLM, such as Vicuna, and fine-tunes the combined system on vision-language instruction data [98]. In the original LLaVA pipeline, the visual encoder remained frozen while the language model was adapted using approximately 158k im- age–instruction examples drawn from MS-COCO and related sources. Practical training techniques—such as gradient checkpointing and data sharding—were employed to reduce GPU memory footprint during fine-tuning. Subsequent LLaVA variants, such as LLaVA-1.5, incorporate larger CLIP ViT backbones and add a small MLP projection layer, improving model capacity with modest adjustments to hyperparameters while maintaining single-image input constraints [99].

Kosmos-1 and Kosmos-2

Kosmos-1 introduced a unified architecture in which multiple input modalities are embedded and directly fed into a causal transformer, enabling the language model to accept both text and image embeddings as native inputs. Training combined large text corpora such as The Pile and Common Crawl with interleaved image–text examples, allowing the model to ground language understanding in visual context. Kosmos-2 extends this approach by incorpo- rating explicitly grounded image–text pairs, enhancing referring and grounding capabilities without relying solely on a two-stage encoder–decoder pipeline. As with many MM-LLMs, CLIP-style image representations serve as the foundation for visual embeddings [100].

MiniGPT-4

MiniGPT-4, released as an open alternative to closed MM-LLMs, demonstrates how a frozen, powerful LLM and a frozen visual encoder can be bridged with a lightweight projection layer to produce robust multimodal behavior. The design keeps both the vision encoder and LLM unchanged during pretraining, training only the projection layer that aligns visual and textual features. A two-stage fine-tuning strategy—first using millions of noisy image–caption pairs, followed by refinement on high-quality image–description samples—significantly improves the generated outputs' coherence and descriptiveness. Empirical studies show MiniGPT-4 outperforming BLIP-2 on creative vision–language tasks such as meme explanation and recipe generation. However, hallucination remains a challenge, particularly for long-form captioning tasks, underscoring the need for balance between model capacity and overfitting control.

mPLUG-OWL

In April 2023, researchers at the Alibaba DAMO Academy introduced mPLUG- OWL, an open-source multimodal large language model (MM-LLM) designed to address key shortcomings in existing two-stage training strategies. Previous MM-LLMs typically relied on fully frozen visual backbones during both pretraining and instruction-tuning phases, which limited the flexibility of cross-modal alignment. To overcome this, mPLUG-OWL proposed a more adaptive approach—retaining trainable visual components in the first stage and freezing them only during the second phase of training.

The model architecture integrates the LLaMA-7B language model (developed in alignment with Vicuna) as the text decoder and a ViT-L/14 vision transformer as the visual encoder. This combination allows mPLUG-OWL to extract detailed visual representations and encode them efficiently as visual tokens. However, integrating such visual features directly into a large language model introduces significant computational challenges due to long input sequences. To address this, the authors introduced a visual abstractor module, which compresses visual embeddings into a compact set of learnable tokens. These condensed visual tokens are then concatenated with the word embeddings of the textual input, ensuring seamless multimodal fusion while maintaining computational efficiency.

The ViT encoder is initialized from CLIP's ViT-L/14 model, leveraging its pretrained weights for faster convergence. During the first stage, the visual encoder and abstractor modules are trained on diverse image–caption datasets, while the language model remains frozen. In the second stage, the focus shifts: the pretrained LLaMA model undergoes LoRA-based fine-tuning (Low-Rank Adaptation) to enhance its ability to interpret text instructions from multiple sources, while the visual modules are frozen. This two-phase structure enables the model to learn effective visual–textual associations and improves generative reasoning across modalities. The LoRA fine-tuning approach is discussed further in Section ??.

To evaluate mPLUG-OWL's performance, the researchers introduced Owl- Eva, a custom evaluation benchmark containing 82 instruction-based questions across 50 images. Results showed that mPLUG-OWL performed competitively against other leading MM-LLMs, including LLaVA, BLIP-2, and MiniGPT-4. Both MiniGPT-4 and mPLUG-OWL demonstrated strong multimodal reasoning and visual comprehension, though mPLUG-OWL exhibited occasional halluci- nation errors, particularly in associating unrelated visual features with textual outputs.

Summary and Comparison of Selected MM-LLMs

A comparative analysis of the reviewed MM-LLMs highlights their distinct strategies for integrating visual and textual modalities. MiniGPT-4 employs frozen vision and language models across both training stages, aligning the modalities through a projection layer that bridges visual and textual features. In contrast, LLaVA maintains frozen vision and language encoders during the initial phase but fine-tunes the language model in the second stage while keeping the vision encoder static.

The mPLUG-OWL framework adopts an inverse training order—its first stage trains the visual encoder and abstractor modules while the language model remains frozen; in the second phase, it fine-tunes the language model (via LoRA) with the vision modules frozen. Meanwhile, Kosmos-1 and Kosmos-2 pursue a single-stage training setup that jointly processes multimodal inputs, using a trainable LLM alongside frozen visual encoders. Taken together, these approaches illustrate that there is currently no consen- sus on the optimal strategy for co-training textual and visual representations in MM-LLMs. Each architecture balances trade-offs between efficiency, generaliza- tion, and alignment accuracy. Figure ?? provides an overview of the respective training paradigms, while Table 2 summarizes their structural and functional characteristics.

Table 2: A Comparative Summary of Selected MM-LLMs

Model	Open Source	Fine-Tuneable	LLM Used	Vision Model Used
LLaVA	Yes	Yes	Vicuna	CLIP ViT-L/14
Kosmos-1 and -2	Yes	Yes	Grounded image–text pairs to train an integrated model	–
MiniGPT-4	Yes	Yes	Vicuna	Q-Former and CLIP ViT-G/14
mPLUG-OWL	Yes	Yes	LLaMA-7B	CLIP ViT-L/14

Model Tuning

Pre-trained Large Language Models (LLMs) and Multimodal Large Language Models (MM-LLMs) hold immense potential across diverse domains. However, their foundational training may not always align perfectly with the specific requirements or contextual nuances of every target application. Certain scenarios may demand more domain-adapted reasoning or task-specific responses that go beyond what is covered during initial pre-training.

To maximise the utility of these foundational models in real-world applica- tions, model tuning techniques are employed. Model tuning enables adaptation of the model's learned parameters or interaction patterns to meet particular goals or contexts. Broadly, model tuning methods can be grouped into four categories: full fine-tuning, parameter-efficient fine-tuning (PEFT), prompt en- gineering, and reinforcement learning with human feedback (RLHF). Each of these techniques serves a distinct purpose, balancing efficiency, adaptability, and resource constraints.

Full Fine-Tuning

Full fine-tuning involves retraining all parameters of a pre-trained foundational model on a smaller, domain-specific dataset. This process tailors the model's generalised knowledge to a specific task, enabling it to better capture the linguistic or multimodal subtleties of that domain.

The key advantage of full fine-tuning lies in its flexibility—it allows the model to comprehensively adjust to new data and objectives, resulting in highly task- aligned outputs. However, the method is computationally demanding and often requires substantial amounts of domain-specific labelled data. Consequently, while it achieves strong performance in specialised tasks, its resource intensity can be a limiting factor [101-102].

Parameter-Efficient Fine-Tuning (PEFT)

Parameter-Efficient Fine-Tuning (PEFT) offers a more practical alternative to full fine-tuning by optimising only a small subset of parameters rather than the entire model. This approach reduces computational cost, memory requirements, and training time while retaining most of the performance benefits.

Since LLMs and MM-LLMs are already trained on large, diverse datasets, they often contain much of the general knowledge necessary for downstream tasks. PEFT capitalises on this by updating only those components relevant to a new objective. Different PEFT methods exist to suit varying needs—some adjust specific sections of the model's parameters, while others introduce lightweight adapter modules that can be trained without altering the base architecture.

Below, we review key PEFT techniques.

Low-Rank Adaptation (LoRA)

Low-Rank Adaptation (LoRA) fine-tunes LLMs or MM-LLMs by freezing the pre- trained model's original weights and inserting small, trainable matrices—known as rank decomposition matrices—into each Transformer layer. This design reduces the number of trainable parameters and memory usage while maintaining strong adaptation capabilities. Once trained, these LoRA adapters can be merged with the original model for inference.
The advantage of LoRA lies in its modularity—multiple LoRA adapters can share the same base model, allowing developers to efficiently manage several task-specific configurations without retraining the entire network.

Quantised Low-Rank Adaptation (QLoRA)

Quantised Low-Rank Adaptation (QLoRA) extends LoRA by introducing quan- tisation, a process that lowers numerical precision to further reduce memory consumption. While LoRA focuses on training compact rank-decomposition matrices, QLoRA additionally applies quantisation to compress model weights, drastically reducing GPU and storage requirements.

This enables the fine-tuning of very large models—up to 65 billion parame- ters—on a single 48GB GPU while maintaining competitive performance. By combining quantisation with low-rank adaptation, QLoRA represents a major step forward in making large-scale fine-tuning more accessible and resource-efficient.

Supervised Fine-Tuning (SFT)

Supervised Fine-Tuning (SFT) uses labelled domain-specific datasets to adapt pre- trained models to specific downstream tasks. Unlike unsupervised pre-training, SFT allows the model to directly learn from human-annotated examples, aligning its outputs with task-specific objectives.

This method enables strong performance with less data and computational demand than training from scratch. However, SFT must be applied care- fully—biases present in the pre-trained or fine-tuning data can become amplified during adaptation. Hence, bias detection and evaluation are essential steps in the SFT pipeline.

Prompt Engineering

Prompt engineering involves crafting natural language instructions—prompts—that guide a model to perform a task without modifying its parameters. By designing effective prompts, models can exhibit in-context learning, where they adapt to new problems simply by interpreting textual cues rather than undergoing further training . This approach mitigates the heavy data and computational requirements of traditional fine-tuning.

Prompting can be categorised into three main types:

- Few-shot prompting, where multiple examples are provided;
- One-shot prompting, where only one example is given;
- Zero-shot prompting, where only the task description is supplied.

Studies suggest that few-shot examples often serve not to teach new tasks, but to help the model locate relevant pre-learned tasks within its latent space. Interestingly, zero-shot performance can sometimes surpass few-shot outcomes. Nevertheless, emphasise that domain-specific prompts—tailored and refined using internal model knowledge—can bridge performance gaps, especially for specialised applications.

Reinforcement Learning with Human Feedback (RLHF)

Reinforcement Learning with Human Feedback (RLHF) enhances model align- ment with human values and preferences. The process begins with human evaluators ranking multiple model-generated outputs. These rankings are then used to train a reward model that estimates the quality of future outputs.

The foundational model is subsequently fine-tuned using reinforcement learn- ing, where the reward model guides it toward producing outputs more consistent with human judgment. While RLHF greatly improves model safety, usability, and alignment, it demands extensive human feedback, data collection, and computational resources—making scalability a significant challenge

Model Evaluation and Benchmarking

Evaluating and benchmarking both pre-trained and fine-tuned models is essential for measuring their capabilities, identifying weaknesses, and assessing the impact of tuning strategies. Before fine-tuning, baseline benchmarks help establish a reference point for performance comparison. After fine-tuning, evaluation metrics assess whether domain adaptation or task specialisation has improved model behaviour.

A notable example is MedAlpaca, a medically fine-tuned LLM evaluated using the USMLE exam, a standardised test for medical practitioners. Its zero-shot results were compared to other models to determine the efficacy of its fine-tuning approach.

Beyond accuracy, evaluation must also consider ethical and social dimensions, including bias, toxicity, and reasoning capability. The RealToxicityPrompts benchmark uses 100,000 prompts to measure a model's likelihood of generat- ing toxic or harmful text via the Perspective API . To detect demographic biases—across gender, religion, race, and socioeconomic attributes—the CrowS- Pairs benchmark compares model perplexity on stereotype versus anti-stereotype sentences Similarly, WinoGender assesses gender bias through co-reference reso- lution tasks, while Winogrande evaluates contextual comprehension by testing pronoun resolution under varying contexts.

Several benchmarks test a model's common-sense reasoning: ARC To evaluate factual accuracy and detect hallucinations, benchmarks such as TruthfulQA and M-HALDetect are employed. TruthfulQA challenges models with 817 diverse questions designed to expose misinformation, while M-HALDetect identifies visual or object hallucinations in multimodal models .

Although these benchmarks provide valuable insights, they serve primarily as indicators rather than guarantees of model safety or fairness. No evaluation framework can conclusively eliminate all risks of bias, hallucination, or misinfor- mation—but comprehensive benchmarking remains the most effective safeguard for developing robust and trustworthy AI systems.

Conclusions

Recent advances in natural language processing have been transformative, par- ticularly with the emergence of Transformer architectures and large language models (LLMs). These models have enabled sophisticated conversational AI systems capable of nuanced reasoning, problem-solving, and contextual understanding. This progress has naturally extended into computer vision, resulting in the development of Multi-Modal Large Language Models (MM-LLMs) and Large Vision-Language Models (LVLMs) such as LLaVA and mPLUG-OWL, which integrate vision encoders with language-based LLMs. Through techniques like fine-tuning and prompt engineering, these models have demonstrated adapt- ability to a variety of domain-specific tasks. Nevertheless, they continue to face persistent challenges, including hallucinations, which, while reducible through strategies such as Visual Contrastive Decoding, cannot yet be entirely eliminated. Open-source MM-LLMs provide distinct advantages in terms of transparency, reproducibility, and control over training data—a critical factor when handling sensitive or proprietary information. Current open-source implementations often do not incorporate the largest or most recent LLMs; for example, MiniGPT-4 and mPLUG-OWL leverage LLaMA-7B and Vicuna-13B, respectively, rather than newer models like LLaMA-2. This choice reflects computational trade- offs associated with advanced techniques such as Reinforcement Learning with Human Feedback (RLHF). The quality and curation of pre-training data are also pivotal, as evidenced by models like Falcon, underscoring that careful dataset
selection remains essential for effective downstream performance.

The usability and performance of MM-LLMs are influenced by multiple factors. Architectural design impacts computational efficiency and the retention of fine- grained visual-linguistic details. Similarly, fine-tuning methods, while beneficial, do not universally guarantee improvements. Comparative studies—such as those between MedAlpaca-13B LoRA and MedAlpaca-13B—highlight that some approaches are too resource-intensive for broad implementation despite their potential performance gains.

While benchmarking provides essential insights into model capabilities and potential risks, no evaluation framework can ensure complete safety, fairness, or elimination of errors. The assessment of domain-specific MM-LLMs requires careful selection of relevant testing procedures, such as the USMLE examination employed to evaluate MedAlpaca's medical reasoning capability. Overall, these considerations are critical for the continued development and evaluation of high- quality MM-LLMs tailored for specialized applications, balancing performance, computational efficiency, and safety in real-world scenarios.

References

1. Ray, P. P. (2023). ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. In Internet of Things and Cyber-Physical Systems. Elsevier.
2. Qu, L., Su, Y., Yu, Z., Lu, P., Yang, S., Zhang, S., & Wang, H. (2023). LayoutLLM-T2I: Eliciting layout guidance from LLM for text-to-image generation. In Proceedings of the 31st ACM International Conference on Multimedia.
3. Qu, L., Su, Y., Yu, Z., Lu, P., Yang, S., Zhang, S., & Wang, H. (2023). LayoutLLM-T2I: Eliciting layout guidance from LLM for text-to-image generation. In Proceedings of the 31st ACM International Conference on Multimedia (MM '23) (pp. 643–654). Association for Computing Machinery. https://doi.org/10.1145/3581783.3612012
4. Lian, L., Chen, Z., Fei, Y., & Xiong, C. (2024). LLM-grounded video diffusion models. In Proceedings of the Twelfth International Conference on Learning Representations. https://openreview.net/forum?id=exKHibougU
5. Ai, Q., Wu, Y., Zhang, M., Mao, J., Xie, X., & Wang, H. (2023). Information retrieval meets large language models: A strategic report from Chinese IR community. AI Open, 4, 80–90. https://doi.org/10.1016/j.aiopen.2023.08.001
6. Lian, L., Chen, Z., Fei, Y., & Xiong, C. (2024). LLM-grounded video diffusion models. In International Conference on Learning Representations (ICLR 2024). https://openreview.net/forum?id=exKHibougU
7. Gupta, P., Ding, B., Guan, C., & Ding, D. (2024). Generative AI: A systematic review using topic modelling techniques. Data and Information Management, 100066. https://doi.org/10.1016/j.dim.2024.100066
8. Anil, R., Chen, X., Dohan, D., Wei, J., Bosma, M., Zhao, V., Chen, Y., Hashimoto, T., & Le, Q. V. (2023). Gemini: A family of highly capable multimodal models. arXiv. https://arxiv.org/abs/2312.11805
9. Zapier. (2024, February 22). Claude 2: A guide to Anthropic's AI model and chatbot. https://www.zapier.com/blog/claude-ai
10. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., & Scialom, T. (2023). LLaMA: Open and efficient foundation language models. arXiv. https://arxiv.org/abs/2302.13971
11. Samsi, S., Thiyagalingam, J., Chenthamarakshan, V., Paluri, M., & Farkas, Z. (2023). From words to watts: Benchmarking the energy costs of large language model inference. In 2023 IEEE High Performance Extreme Computing Conference (HPEC) (pp. 1–9). IEEE. https://doi.org/10.1109/HPEC58863.2023.10255464.
12. Rillig, M. C., Ryo, M., Lehmann, A., Wang, J., Camenzind, T., & Grilli, G. (2023). Risks and benefits of large language models for the environment. Environmental Science & Technology, 57(9), 3464–3466. https://doi.org/10.1021/acs.est.2c09203
13. Tokayev, K.-J. (2023). Ethical implications of large language models: A multidimensional exploration of societal, economic, and technical concerns. International Journal of Social Analytics, 8(9), 17–33.
14. Barney, N. (2024). What is language modeling? — Definition from TechTarget. https://www.techtarget.com/searchenterpriseai/definition/language-modeling
15. Drost, D. (2023). A brief history of language models. Towards Data Science. https://towardsdatascience.com/a-brief-history-of-language-models-d9e4620e025b
16. Anandika, A., Mishra, S. P., & Das, M. (2021). Review on usage of Hidden Markov Model in natural language processing. In D. K. Behera & J. Nayak (Eds.), Intelligent and cloud computing: Proceedings of ICICC 2019 (Vol. 1) (pp. 415–423). Springer.
17. Castilho, S., Shterionov, D., Trushkina, J., Espaillat, A., & Way, A. (2017). Is neural machine translation the new state of the art? The Prague Bulletin of Mathematical Linguistics, 108, 109–120. https://doi.org/10.1515/pralin-2017-0013.
18. Patil, R., Boit, S., Gudivada, V., & Nandigam, J. (2023). A survey of text representation and embedding techniques in NLP. IEEE Access. https://doi.org/10.1109/ACCESS
19. Cui, C., Chen, H., Wei, X., Huang, S., Xu, Z., & Wang, L. (2023). A survey on multimodal large language models for autonomous driving. arXiv. https://arxiv.org/abs/2311.12320
20. McCloskey, B. (2024, February 22). Choosing neural networks over n-gram models for natural language processing. Towards Data Science.
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. arXiv. https://arxiv.org/abs/1706.03762.
22. Huang, S., Wang, X., Dong, L., Wei, F., & Wang, L. (2023). Language is not all you need: Aligning perception with language models. arXiv. https://arxiv.org/abs/2302.14045.
23. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv. https://arxiv.org/abs/1810.04805
24. Marr, B. (2023). A short history of ChatGPT: How we got to where we are today. Forbes. https://www.forbes.com/sites/bernardmarr/2023/05/19/a-short-history-of-chatgpt-how-we-got-to-where-we-are-today/
25. Boesch, G. (2024, March 7). Llama 2: The next revolution in AI language models — Complete 2024 guide. viso.ai. https://viso.ai/deep-learning/llama-2/
26. Agarwal, S. (2024, February 23). Navigating the attention landscape: MHA, MQA, and GQA decoded. Medium. https://iamshobhitagarwal.medium.com/navigating-the-attention-landscape-mha-mqa-and-gqa-decoded-288217d0a7d1
27. Ainslie, J., Ontañón, S., Alberti, C., Lazaridou, A., & Tay, Y. (2023). GQA: Training generalized multi-query transformer models from multi-head checkpoints. arXiv. https://arxiv.org/abs/2305.13245.
28. Chanen, A. (2023). What I learned from Bloomberg's expe-

rience of building their own LLM. LinkedIn. https://www.linkedin.com

29. Guaglione, S. (2023). The case for and against open-source large language models for use in newsrooms. Digiday. https://digiday.com
30. IBM Data and AI Team. (2024). Open source large language models: Benefits, risks and types. IBM. https://www.ibm.com
31. Khudov, N. (2023). The future of LLMs: Proprietary versus open-source. LinkedIn. https://www.linkedin.com
32. Streng, S. (2023). Game-Changer 2024: Meta's LLAMA 2.0. Medium. https://medium.com/@sebastianstreng96/game-changer-2024-metas-llama-2-0-4ab1316b6aa4
33. European Parliament. (2024, March 15). EU AI Act: First regulation on artificial intelligence. https://europarl.europa.eu
34. Pandiyan, I. (2023). Open source or proprietary LLMs. Medium. https://medium.com
35. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … & Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901.
36. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8), 1–9.
37. Achiam, J., Adler, S., Agarwal, S., Arora, S., Barnes, E., Barron, D., Blackwell, S., Brundage, M., Carver, B., Clark, J., & Kaplan, J. (2023). GPT-4 technical report. arXiv. https://arxiv.org/abs/2303.08774. Anthropic. (2024, March 26). Introducing Claude. https://www.anthropic.com
38. Yue, X., Hu, S., Chen, X., Chen, W., Zhou, Y., Liu, Z., & Zhao, H. (2023). MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. arXiv. https://arxiv.org/abs/2311.16502.
39. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., & Scialom, T. (2023). LLaMA: Open and efficient foundation language models. arXiv. https://arxiv.org/abs/2302.13971
40. Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, H. F., Henighan, T., Reynolds, L., Bouchard, G., Joulin, A., Aitchison, M., Lerer, A., & Sifre, L. (2022). Scaling language models: Methods, analysis & insights from training Gopher. arXiv. https://arxiv.org/abs/2112.11446
41. Hoffmann, J., Borgeaud, S., Mensch, A., Millican, K., Cai, T., Rutherford, E., Driessche, G. V., Hendricks, L. A., Ring, R., Stoker, J., & Irving, G. (2022). Training compute-optimal large language models. arXiv. https://arxiv.org/abs/2203.15556
42. Tsang, S.-H. (2024, March 5). Review: LLaMA — Open and efficient foundation language models. Medium. https://medium.com
43. Johnson, A. (2023). Understanding rotary position embedding: A key concept in transformer models. Medium. https://medium.com
44. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., & Synnaeve, G. (2023). LLaMA 2: Open foundation and fine-tuned chat models. arXiv. https://arxiv.org/abs/2307.09288.
45. Patel, A. A. (2023). LLaMA 1 vs LLaMA 2: A deep dive into Meta's LLMs. Ankur's Newsletter. https://www.ankursnewsletter.com/p/llama-1-vs-llama-2-a-deep-dive-into
46. Walker, S. M. (2024, March 7). What is grouped query attention (GQA)? Klu.ai. https://klu.ai/glossary/grouped-query-attention
47. Pierre, R. (2024, March 8). Parameter-efficient fine-tuning (PEFT): Enhancing large language models with minimal costs. MLOpsHowTo. https://mlopshowto.com
48. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, L., Wang, X., & Chen, W. (2021). LoRA: Low-rank adaptation of large language models. arXiv. https://arxiv.org/abs/2106.09685.
49. Han, T., Schuster, T., He, J., Zbib, R., Chheda, H., Patil, D. J., & Alon, U. (2023). MedAlpaca: An open-source collection of medical conversational AI models and training data. arXiv. https://arxiv.org/abs/2304.08247.
50. Child, R., Gray, S., Radford, A., & Sutskever, I. (2019). Generating long sequences with sparse transformers. arXiv. https://arxiv.org/abs/1904.10509
51. Technology Innovation Institute. (2024). UAE's Technology Innovation Institute launches open-source Falcon 40B large language model for research commercial utilization. TII. https://www.tii.ae
52. Hoque, M. (2023). Exploring the Falcon LLM: The new king of the jungle. Medium. https://medium.com/@minh.hoque/exploring-the-falcon-llm-the-new-king-of-the-jungle-5c6a15b91159
53. Werra, L. de, Wolf, T., Schmid, P., Lozhkov, A., Rae, J. W., & Berman, S. (2023). The Falcon has landed in the Hugging Face ecosystem. Hugging Face. https://huggingface.co/blog/falcon.
54. Almazrouei, E., Rozière, B., Abdalla, M., Mazzawi, H., Boral, A., Elhussein, M., Lialin, V., Maguire, A., & Stojnic, R. (2023). The Falcon series of open language models. arXiv. https://arxiv.org/abs/2311.16867.
55. Penedo, G., Malartic, Q., Graciosa, M., Kaddar, Y., Vaswani, A., Habib, A., & Azhar, F. (2023). The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. arXiv. https://arxiv.org/abs/2306.01116.
56. Mukherjee, S. (2023). Introducing Falcon 180B: A comprehensive guide with a hands-on demo of the Falcon 40B. Paperspace. https://blog.paperspace.com/introducing-falcon/
57. Oyerinde, F. (2023). BLIP-2: A breakthrough approach in vision-language pre-training. Medium. https://medium.com/@femiloyeseun/blip-2-a-breakthrough-approach-in-vision-language-pre-training-1de47b54f13a
58. Li, J., Li, D., Savarese, S., & Hoi, S. (2023). BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv. https://arxiv.org/abs/2301.12597
59. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., & Houlsby, N. (2020). An image is worth 16×16 words: Transformers for image recognition at scale. arXiv. https://arxiv.org/abs/2010.11929.
60. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 770–778).
61. Li, Y., Mao, C., Garg, S., Wu, B., Pei, K., Chen, C., & Song, D.

(2023). Evaluating object hallucination in large vision-language models. arXiv. https://arxiv.org/abs/2305.10355

62. Ghandi, T., Pourreza, H., & Mahyar, H. (2023). Deep learning approaches on image captioning: A review. ACM Computing Surveys, 56(3), 1–39. https://doi.org/10.1145/3594511
63. Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2023). Visual instruction tuning. arXiv:2304.08485
64. Xie, C.-W., Zhang, R., Yu, Q., Li, Z., & Huang, G. (2023). RA-CLIP: Retrieval-augmented contrastive language-image pre-training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 19265–19274). https://doi.org/10.1109/CVPR52729.2023.01846
65. Chiang, W.-L., Li, Z., Lin, H., Sheng, Y., Wu, Z., Zhuang, S., Li, Z., & Zhu, X. (2023). Vicuna: An open-source chatbot impressing GPT-4 with 90% ChatGPT quality. LMSYS Blog. https://lmsys.org/blog/2023-03-30-vicuna/.
66. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), ECCV 2014: 13th European Conference on Computer Vision. Springer, 740–755.
67. Liu, H., Li, C., Li, Y., & Lee, Y. J. (2023). Improved baselines with visual instruction tuning. arXiv:2310.03744.
68. Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presani, E., & Leahy, C. (2020). The Pile: An 800GB dataset of diverse text for language modeling. arXiv. https://arxiv.org/abs/2101.00027.
69. Common Crawl Foundation. (2024). Common Crawl. https://commoncrawl.org/
70. Peng, Z., Li, Z., Du, Y., Zhang, J., Yang, J., Wang, C., & Liu, Z. (2023). Kosmos-2: Grounding multimodal large language models to the world. arXiv. https://arxiv.org/abs/2306.14824.
71. Zhu, D., Chen, W., Shen, Y., Li, K., Cao, Z., Li, C., & Keutzer, K. (2023). MiniGPT-4: Enhancing vision-language understanding with advanced large language models. arXiv. https://arxiv.org/abs/2304.10592.
72. Chiang, W.-L., Li, Z., Lin, H., Sheng, Y., Wu, Z., Zhuang, S., & Zhu, X. (2023). Vicuna: An open-source chatbot impressing GPT-4 with 90% ChatGPT quality. LMSYS Blog. https://lmsys.org/blog/2023-03-30-vicuna/. (Accessed February 15, 2024)
73. Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning (ICML) (pp. 8748–8763). PMLR.
74. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding with unsupervised learning (Technical report). OpenAI.
75. Moslem, Y., Haque, R., Kelleher, J. D., & Way, A. (2023). Adaptive machine translation with large language models. In Proceedings of the 24th Annual Conference of the European Association for Machine Translation (pp. 227–237). https://aclanthology.org/2023.eamt-1.22
76. Yang, J., Yu, Z., Wang, Z., He, J., Chen, X., & Zhang, Y. (2024). Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond. ACM Transactions on Knowledge Discovery from Data. https://doi.org/10.1145/3649506
77. Raiaan, M. A. K., Ahmed, M., Rahman, M. H., Islam, M. M., & Rahman, M. S. (2024). A review on large language models: Architectures, applications, taxonomies, open issues and challenges. IEEE Access, 12, 26839–26874. https://doi.org/10.1109/ACCESS.2024.3365742
78. Patel, D. (2023). Google "We have no moat, and neither does OpenAI". SemiAnalysis. https://semianalysis.com
79. Ye, Q., Wang, B., Yang, Q., Liu, Y., Chen, L., & Huang, F. (2023). mPLUG-Owl: Modularization empowers large language models with multimodality. arXiv. https://arxiv.org/abs/2304.14178.
80. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., …& Houlsby, N. (2021). An image is worth 16×16 words: Transformers for image recognition at scale. arXiv. https://arxiv.org/abs/2010.11929.
81. Nabwani, N. (2023). Full fine-tuning, PEFT, prompt engineering, or RAG? Deci.ai. https://deci.ai/blog/fine-tuning-peft-prompt-engineering-and-rag-which-one-is-right-for-you (Accessed February 17, 2024)
82. Xu, L., Cai, Y., Xu, H., Zhu, K., Wang, Y., Xu, R., & Wang, B. (2023). Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. arXiv. https://arxiv.org/abs/2312.12148.
83. Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). QLoRA: Efficient finetuning of quantized LLMs. arXiv:2305.14314.
84. Martinez, J. J. (2023). Supervised fine-tuning: Customizing LLMs. Medium. https://medium.com/mantisnlp/supervised-fine-tuning-customizing-llms-a2c1edbf22c3 (Accessed March 2, 2024)
85. Norouzi, A. (2023). The ultimate guide to LLM fine-tuning: Best practices & tools. Lakera. https://www.lakera.ai/blog/llm-fine-tuning-guide (Accessed March 2, 2024)
86. Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys, 55(9). https://doi.org/10.1145/3560815.
87. Garg, S., Tsipras, D., Liang, P. S., & Valiant, G. (2022). What can transformers learn in-context? A case study of simple function classes. In Advances in Neural Information Processing Systems (Vol. 35, pp. 30583–30598).
88. Reynolds, L., & McDonell, K. (2021). Prompt programming for large language models: Beyond the few-shot paradigm. In Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1–7).
89. Sun, L., Wang, L., Sun, J., & Okatani, T. (2023). Prompt prototype learning based on ranking instruction for few-shot visual tasks. In 2023 IEEE International Conference on Image Processing (ICIP) (pp. 3235–3239). https://doi.org/10.1109/ICIP49359.2023.10222039
90. Now Next Later AI. (2023). What is RLHF: Reinforcement learning from human feedback. Medium. https://medium.com/generative-ai-insights-for-business-leaders-and/what-is-rlhf-reinforcement-learning-from-human-feedback-876da930bf16 (Accessed March 2, 2024)
91. Gehman, S., Gururangan, S., Zellers, R., Choi, Y., & Smith, N. A. (2020). RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In Findings of the Association for Computational Linguistics: EMNLP 2020. Asso-

ciation for Computational Linguistics, 3356–3369. https://doi.org/10.18653/v1/2020.findings-emnlp.301.

92. Lees, A., Sorensen, T., Tsvetkov, Y., Hutchinson, B., Pavlick, E., & Dixon, L. (2022). A new generation of Perspective API: Efficient multilingual character-level transformers. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22) (pp. 3197–3207). ACM. https://doi.org/10.1145/3534678.3539147
93. Nest, D. (2023). LLM benchmarks: What do they all mean? WhyTryAI. https://www.whytryai.com/p/llm-benchmarks (Accessed March 2, 2024)
94. Clark, P., Khot, T., Sabharwal, A., Tafjord, O., Clark, C., & Etzioni, O. (2018). Think you have solved question answering? Try ARC, the AI2 reasoning challenge. arXiv. https://arxiv.org/abs/1803.05457.
95. Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., & Choi, Y. (2019). HellaSwag: Can a machine really finish your sentence? arXiv. https://arxiv.org/abs/1905.07830.
96. Clark, C., Lee, K., Chang, M.-W., Kwiatkowski, T., Collins, M., & Toutanova, K. (2019). BoolQ: Exploring the surprising difficulty of natural yes/no questions. arXiv. https://arxiv.org/abs/1905.10044.
97. Mihaylov, T., Clark, P., Khot, T., & Sabharwal, A. (2018). Can a suit of armor conduct electricity? A new dataset for open book question answering. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2381–2391.
98. Bisk, Y., Zellers, R., Gao, J., & Choi, Y. (2020). PIQA: Reasoning about physical commonsense in natural language. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 05, pp. 7432–7439).
99. Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., & Steinhardt, J. (2020). Measuring massive multitask language understanding. arXiv. https://arxiv.org/abs/2009.03300.
100. Lin, S., Hilton, J., & Evans, O. (2022). TruthfulQA: Measuring how models mimic human falsehoods. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 1, 3214–3252.
101. Gunjal, A., Yin, J., & Bas, E. (2023). Detecting and preventing hallucinations in large vision language models. arXiv:2308.06394.
102. Mashette, N. (2023). LLM benchmarks (Introduction to benchmark techniques). Medium. https://medium.com/@nageshmashette32/llm-benchmarks-introduction-to-benchmarks-techniques-6518527620eb. (Accessed March 2, 2024)